1992

Simulation of human preattentive mechanism for detection of the area of interest

Narendra A. Kulkarni
University of Wollongong

Simulation Of Human Preattentive Mechanism For Detection Of The Area Of Interest

A thesis submitted in fulfilment of the requirements

for the award of the degree

MASTER OF ENGINEERING

from

University of Wollongong

by

Narendra A. Kulkarni, M. Sc.

Department of Electrical and Computer Engineering

June, 1992

Acknowledgements

"If you shut your door
to all errors
truth will be
shut out"
– Rabindranath Tagore –

On coming back to the academic field after a long time, I realized that I needed some time to adjust to the vagaries of academic freedom from the rote of Industrial life. My supervisor **Dr. Golshah A. Naghdy** has been remarkably supportive in my efforts to adjust to the new environment. She provided the necessary guidance as and when required, and made valuable suggestions. I wish to thank her for helping me to develop this ability, to undertake independent research work and, allowing me the privilege of working with her on this project. I would also like to thank her for the time and effort she put in during this research.

Many people have contributed to the completion of this project and, like as I may, to thank all of them, I might miss out a few. I wish to thank **Prof. Chris Cook** for giving me this opportunity to work unhindered on my research project and supporting me with a fellowship without which this endeavor would have been difficult to undertake. I wish to thank **Dr. Fazel Naghdy**, the graduate Coordinator for providing the facilities necessary for my research work.

My completing this project in the UNIX, X Windows environment on the Sun Network of the campus would have been near impossible without the active support of **Dr. Phillip Ogunbona**. Others who helped me in coping with the idiosyncrasies of the Sun Workstation are Van dao Mai, Kirk Barrett, Ian Piper and Minh Luu of the ITS.

I wish to thank **Maree Fryer** without whom I would not be in Australia now to complete this project. She has been of immense help to me in managing all those administrative problems which would have been a hindrance without her.

I would also like to thank staff members of the computer science department espe-

cially **David Wilson, Peter Gray** and **Steve Cliffe** for extending all the support to me in times of difficulty.

This effort would have been lacking a firm mathematical background without the insight provided by **Dr. R. V. Nillsen** of the Mathematics Dept. After attending his lectures on *Measure and Integration* I could think of tackling the problem in a more systematic, well defined manner.

A special vote of thanks to **Juha Tuominen** for introducing me to the powerful text formatting of TEX/ LATEX.

Finally I would like to thank Tracey King, Vesna Gospic, Peter Costigan, Carlo Giusti, Kan Kandasamy, all the staff members and my colleagues in the department for helping me as and when necessary.

It would be highly remiss on my part, if I do not mention that, I have the solid support of my **parents and uncles**, without which, I could not have persued my ambition of further studies in the field of Computer Engineering.

Naren Kulkarni
WOLLONGONG
June 1992

Contents

List of Figures

Chapter 1

INTRODUCTION

"The proper means of increasing the love we bear our native country is to reside some time in a foreign one".

– WILLIAM SHENSTONE 1714-1763 –

1.1 Background

The most important sensory perception mechanism that may be employed for a robotic system is that of *Vision.* Almost since the time when digital computers (Von Neumann machines) became available there have been concerted efforts to impart the faculty of vision to machines [Bro86], [KPH86], [GN89].

As is true in humans, vision in a robot facilitates a sophisticated sensing mechanism that allows the machine to respond to its environment in an intelligent and *adaptable* manner.

Vision in humans is apparently an instantaneous and effortless event. One need only open one's eyes to see and sense the surroundings. This ease of the human visual mechanism belies the underlying sophistication and precision involved in collecting the image information and the complexities involved in processing it with apparent ease [LYU90].

Vision is almost as complicated as it is powerful. Knowledge about biological vision systems is fragmented and, it is very difficult to implement a fully operational machine vision system. In order to get a working machine vision system one has to understand the higher level functions carried out by the biological vision systems — perception,

cognition and pattern analysis.

Implementation of machine vision calls for a thorough understanding of the process of image formation.

Two basic approaches may be considered in the implementation of machine vision systems.

- Development of an algorithm for implementing the *task-at-hand.*
- Emulation of existing biological vision systems.

The approach adopted here, tends more towards emulation of existing biological systems.

The state of the art *vision systems* available today cannot be utilized for universal applications. Most of the machine vision systems today address a particular task that is carried out in a well defined environment [GN89]. Most of the progress has been made in the field of industrial applications where the visual environment can be controlled and the job at hand for the machine vision system is totally defined. For example recognizing and picking up machine parts off an assembly line or conveyor belt [Bro86] and [KPH86].

Navigating a mobile robot presents a different kind of challenge. A robot may be trained to chart its path around a given factory shop floor. Enabling it to move around unforeseen obstacles is much more complex especially when the path charting algorithm assumes that the path is clear of obstacles.

Robot vision systems have certain special considerations. Huge amounts of image data has to be handled and analysed by the robot vision system in *real-time.* There is a need to reduce the amount of data to be processed by the robot vision system for comprehension. The solution must incorporate some sort of preprocessing which gets rid of most of the unwanted data and retains data pertaining to the *"areas of interest".*

Machine vision systems or any vision system for that matter may be looked upon as an element of a feedback path concerned with sensing, where other elements, like the brain in the *Human Visual System,* and the CPU of the computer in case of machine vision, are delegated other tasks like decision making and implementing those decisions. The human visual system utilises the fedback it gets from the brain for the

purpose of concentrating on the area of interest and getting rid of the unwanted area.

The mechanism that allows the human visual system to take in the surroundings without focussing on any particular object, the moment the eyes are opened is termed as *Preattentive mechanism* [Nag90].

For any task-at-hand there will be an *"area of interest"* from the total field of view. Human preattentive mechanism allows the visual system to get rid of the unwanted areas. The *"attentive"* visual system can then concentrate on the area of interest to accomplish the task-at-hand.

The human visual system scans the entire field of view without any particular part of the view in focus to take in the general state of surroundings. When the visual system concentrates on a particular object within the field of view, the area of *field of view* comprising the object is scanned at a higher resolution (optimum required level) to register the details of that object. The distinction between preattentive and attentive mechanisms is not simple to make. The area of interest in the entire field of view can be segregated only if it differs from the rest of the field in certain features that are detected by the the visual system.

The human preattentive mechanism may be adopted for simulation as a feature detection mechanism in machine vision system.

The human visual system has detection mechanisms for features like **colour, orientation**, and channels tuned to particular **frequencies**.

The cells tuned to detect different features fire as and when they receive stimulus of the feature to which they are tuned, during the *preattentive* phase. This gives an overall image of the surroundings. The eye then scans the area of interest at optimal resolution when attention is focussed on it in the *fixation* phase.

The human visual system takes into account the variations in spatial frequency, and those cortical cells that are tuned for particular frequencies and orientation fire when they are presented with a stimulus that corresponds to that frequency and orientation.

In this project the concentration is on the feature *"frequency"* i.e., spatial-frequency of the image signal in two dimensions, as a mechanism for the detection and segregation of the area of interest. Spatial-frequency translates to mean texture in terms of

spatial grain size. The relationship between spatial grain size and spatial frequency is as described in section 4.6.2.1.

In order to simulate the human visual system, the basic functioning of simple cell in the human cortex needs to be modelled or simulated. It has been shown that Gabor Elementary functions [GD88] are very good estimations for the *receptive field profiles* of the simple cortical cells tuned to a specific frequency and orientation.

Segmentation is the process that subdivides a sensed scene into its constituent parts or objects. Segmentation is essential for the functioning of a machine vision system, since segmentation leads to picking out of an object from a cluttered scene and using it for further cognition and analysis.

1.1.1 Spatial Resolution in man and machine

In order to segment texture of interest, it is necessary to determine the measure of resolution since spatial resolution in man and machine are two intrinsically different quantities.

The human eye can focus from a distance of *25 cms.* to infinity, and spatial resolution is essentially an angular quantity as far as the eye is concerned. Spatial resolution may be specified in the spatial domain as the ability of the eye to sense the separation in two dots that have been placed close together - or it may be specified as a variation in attenuation of the amplitudes of intensity in the image, in the spatial frequency domain.

The spacing between two samples (pixels) of the image for machine processing must be such that it satisfies the Nyquist criterion of the sample spacing, that "Δx" the sample spacing between two pixels, must be less than the total sample space "d" divided by $2f$ where f is the highest spatial frequency at which information or energy is present in the image. Spatial resolution for machine vision is defined in terms of cycles/sample i.e. changes in light intensity per sample space. This sampling rate is also known as the bit-rate in one dimensional signals that have time as the independant parameter. For machine processing of an image, the ideal format of sampling would be a hexagonal grid of picture elements (pixels) such that each pixel is equidistant from six of its nearest neighbours. But since it is very difficult to generate an image pattern in a hexagonal fashion, it is convenient to have square grids of size $(m \times n)$ pixels where m, n are often even numbers and $m = n$ [Nib86].

1.2 Objective

The primary objectives of this project are the detection of text matter from a scene containing text with natural surroundings, i.e. filtering the unwanted information from the scene, and focusing on the area of interest.

In this project it is proposed to use a single low resolution video camera, fitted onto a mobile robot navigation system, to scan a restricted environment such as a factory shop floor or work shop. The scanned image is to be processed so that the system can isolate different texture patterns within a radius of $1.5 - 2$ metres and segment out i.e. segregate the region of interest. This has the effect of *Localizing* the area of interest to a small part, or the aggregate of small parts, of the entire image - i.e. *fixation* for machine vision. Once the area of interest has been segmented, that area can be scanned at the optimum resolution in order to utilize the information therein efficiently and effectively.

This project describes an algorithm to distinguish between different textures. It performs the filtering in spatial domain in a single operation, thus reducing the computational overhead.

1.3 Applications

This project has the following applications:

1. Segmentation of regions of interest including *text matter* and/or labels or directions written on the walls, from the natural background for a mobile, vision guided navigation system.

   [-] The above mentioned application finds widespread use in the field of robotics and business machines like scanners and photocopiers i.e. in the field of document imaging.

2. Another possible application of this spatial/spatial-frequency based approach to image filtering and simulation of human vision system is in the field of psychology as detailed in Appendix D. This application was not tried out in the process of this project, but the psychology department has been interested in this approach

to solving the problem of Dyslexia due to visual impairment.

1.4 Overview

The similarities and the differences in human and machine vision systems are discussed in Chapter 2. The concept of low-level vision and some models for the simulation of the simple cortical cell are also discussed.

In order to get an image in a matrix form mentioned above, i.e., a form amenable for machine processing, one has to scan the scene with a camera that will digitize the whole image at a specified sampling rate. This mechanism of discretization of the image using different means like vidicon and/or the CCD camera is discussed at length in Chapter 3. The hardware equipment that comes into play for the purpose of image grabbing is also discussed.

Once the image has been discretized, one should take a look at the various methods involved in image processing for computer vision. Segmentation techniques based on edge detection, and textural features lay the foundation for a spatial/spatial-frequency (s/sf) representation of an image and are discussed in Chapter 4.

The uncertainty principle that lays down the limitations of an image representation, with respect to resolution, in the conjoint spatial/spatial-frequency (s/sf) domain are also discussed herein. This leads to the development of the short time Fourier transform or the window Fourier transform.

The window Fourier transform is widely used in speech signal processing and other communication applications involving analysis of statistically non-stationary signals. This representation has found application in recent times in the field of machine vision due to its property of segregating textural variations that are localized spatially. A special type of window Fourier transform, the *Gabor Transform*, which uses a Gaussian window that has the property of achieving the lower bound of resolution entropy in the conjoint spatial/spatial-frequency domain is discussed in Chapter 5. A systematic analysis of the problem involved in this project i.e. texture segmentation, for multiresolution analysis of an image, is presented along with the mathematical formulation, which enables the study of this problem on a formal basis.

Implementation and application of the Gabor Transform to image analysis is ex-

plained in Chapter 6, and some of the applications that have been presented in this chapter, lead to a conclusion that Gabor filter functions may be used for localizing the texture, and segmenting regions having similar texture. The case wherein two textures display spatial frequencies that are close to each other is presented, wherein the Gabor functions using the Gaussian Window give an overlapping effect, demonstrating the constraints over frequency selectivity of the Gabor filter operator. The relationship between the *"spread"* of the gaussian window in the frequency domain and the effective resolution obtained in the spatial domain are discussed.

In "Conclusion" the author has presented the comparative advantages and drawbacks of using the Gabor filters as a tool for feature detection visavis other *spatial/spatial-frequency* representations.

On this background other options like the wavelet theory, and its ramifications in the field of texture/feature analysis as compared with the Gabor functions are also discussed. The recovery of the original signal from its Gabor transform, on the basis that it may be considered as a special (orthonormal) case of the *Wavelet* is also discussed.

The author has presented background material necessary for explaining some of the topics in *Appendices* A, B, C and D.

Appendix A discusses the biological functioning of the human eye and the eye-brain system.

Appendix B discusses in brief the mathematical background necessary for signal theory and analysis.

Appendix C details the procedures adopted during the project for manipulating images.

Appendix D discusses the possible application of the basic principles of psychology involved in this project regarding people suffering from the effects of **Dyslexia**.

Chapter 2

THE LINKS IN HUMAN AND MACHINE VISION

" In every work of genius, we recognize
our own rejected thoughts;
they come back to us with a certain alienated majesty".
– Ralph Waldo Emerson –

Whether one implements a machine vision system that serves a restricted purpose and does not resemble the human visual system in any of its aspects, or emulates the human visual system - the basic requirements for a machine vision system are the same:

- Image Capture - formation of the discrete image that is useful for machine processing, from the scene to be analyzed.
- Image Analysis - applying various techniques for processing the image - conversion to a form that is easier to analyze.
- Segmentation - segregating a given image frame into its individual components which then may be studied and/or analyzed separately if necessary.
- Decision making. - the function performed by the brain in case of the human visual system.

Most approaches in computer vision are derived from some similar feature of the human visual System. It will be of immense help if all the findings in the field of human

visual system can be explained in terms of some aspect of computer vision. It would enable one to better understand the human visual system [RRW90].

Since it is proposed to emulate some of the features of the human visual system, it would be instructive to study the functioning of the human visual system in some detail.

2.1 Human visual system

In the human eye, the cells of the cortex in and around the *Fovea Centralis* generate impulses on sensing light intensity. The frequency of the generated impulses is less than 1000 Hz. The number of impulses in a burst is proportional to the rate of change of light intensity with respect to time. Rapid but small movements of the eye (saccadic movements) make sure that a still image (picture) also produces a time varying stimulus, thus generating bursts of impulses [Bro86].

The total field-of-view of a human eye[1] may extend to 60 degrees on both sides but the image is processed at the highest resolution only around the fovea centralis. The resolution decreases on all sides of the fovea centralis with increasing angle from the visual axis [Bro86].

The fovea centralis, which is at the center of the eyeball, processes the image at a high resolution and is sensitive to red and green colour since it does not contain blue cones. As one might observe, from the center of the fovea to the periphery of the cortex the cells of the eye become sensitive to phase, orientation, contrast, and all three basic colours, i.e. red, green and blue, but they process the image information at a much lower resolution. This is because there are two kinds of receptors in the visual system, namely rods and cones. The center of the fovea, which subtends a very narrow angle on the field of view contains only cones. The number of cones falls off rapidly with increasing distance from the center of the fovea. The retina does not have an even distribution of the receptor cells across its breadth. The fovea being packed with cones gives the best resolution, but it is less sensitive than the surrounding regions that contain the rods.

The human visual system is *totally non-linear*. When a human observer is scanning an image, the eye moves so as to scan small parts of the image, which contain the

[1]Refer to index A for the working of the eye-brain system

information necessary to further the viewer's analysis of the image, with its foveal region. Hence the part of the image that is scanned repeatedly depends upon what the observer is looking for. A particular region may be scanned many times before proceeding to analyze the scene. This facility of small rapid movements in the human eye helps overcome the aberrations in eye lens, since a sharp image can be obtained in a very narrow angle, around the optic axis.

Thus the scanning process is very efficient and economical — any information that is not needed by the human perceiver is discarded and not recorded by the eye and brain combination. Since one cannot emulate the saccadic movements of the eye due to constraints on the amount of data that may be processed in real-time, it is better to emulate the preattention mechanism using spatial filtering techniques.

2.1.1 Low level vision

In the human visual system as the image information is processed in real-time. This primary processing is low level vision.

It is difficult to put down a definitive description of what constitutes low level vision. Low level vision is concerned with the sensing of physical properties of the surrounding visual environment, such as the depth of various points within the field-of- view, orientation, material textures, motion and edge detection [LYU90].

Many of the processes connected with low level vision take place in parallel. Edge detection, texture boundary sensing, depth sensing from binocular vision, shape sensing from texture and contours, all take place in parallel.

The mechanism which takes in the image and processes it in parallel does not really focus the eye on any object. Focussing takes place only after the decision that a particular object is of interest has been made. Picking out an object of interest from a field of view, and then focussing on it, is termed *fixation.*

2.1.2 Model of a simple retinal cell

In this project it is proposed to simulate the functioning of a simple cortical cell. The *receptive field profile* of a simple cortical cell is to be to simulated under the condition of stimulation due to any image data function. A simple cortical cell is presumed to get its inputs from the summation of the outputs of center-surround units of the *lateral geniculate nucleus* LGN [GD80].

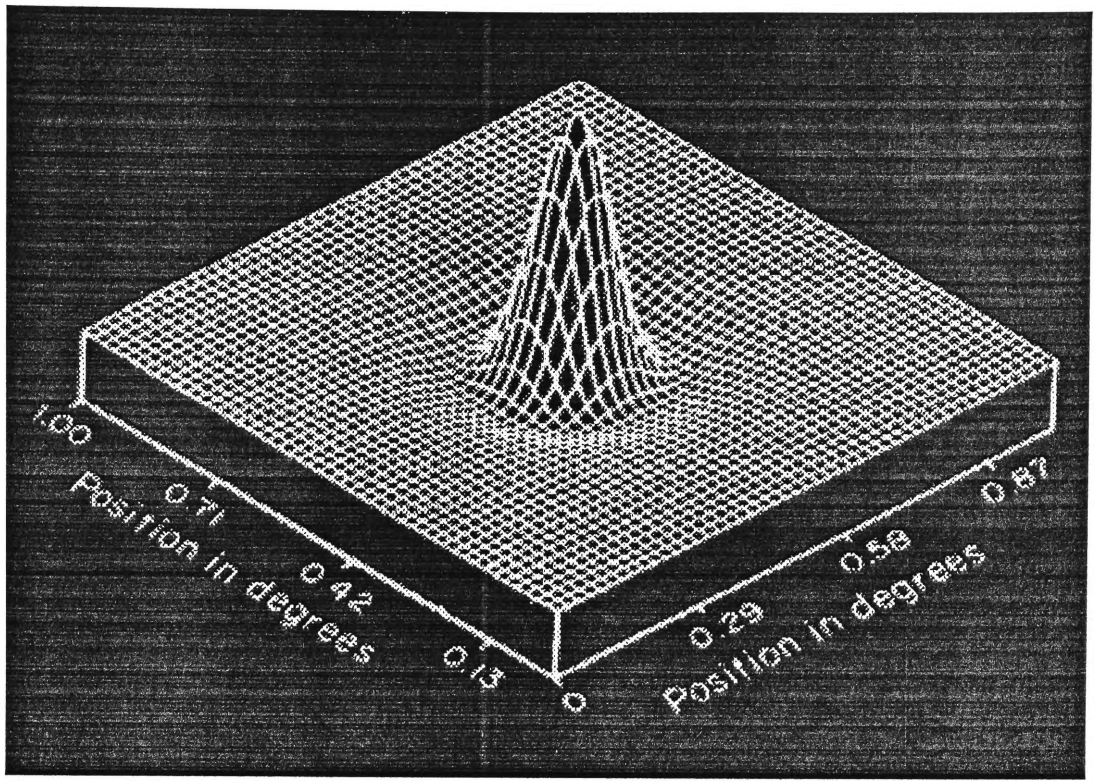

Figure 2.1: DOG representation of a single LGN cell rfp

A simple cell's receptive field profile is essentially a mathematical representation of a cell's response characteristics in terms of a function $g(x, y)$, that is defined over the visual space (x, y).

The function $g(x, y)$ on multiplication by another function $f(x, y)$ that defines the distribution of light intensity as a stimulus function, and then its integration over the (x, y) plane, gives the response of the simple cell to the stimulus. This response function may be defined by [GD80]

$$response = k \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) g(x, y) dx\, dy.$$

In the above response function one has to assume linearity. i.e. scaling of the stimulus function $f(x, y)$ should also scale the response in similar fashion.

The basic logic for modeling the *orientation selective receptive fields*, can be traced back to the qualitative notion originally advanced by Hubel and Weisel (1962), and quoted by Daugman [GD80]. It states that elongated cortical cell profiles are made of an aligned row of inputs from summated center-surround LGN cells. The weighting function of each LGN component was later on characterized [GD80][2] to be the

[2]This depiction of LGN cells is as shown by Daugman [GD80]

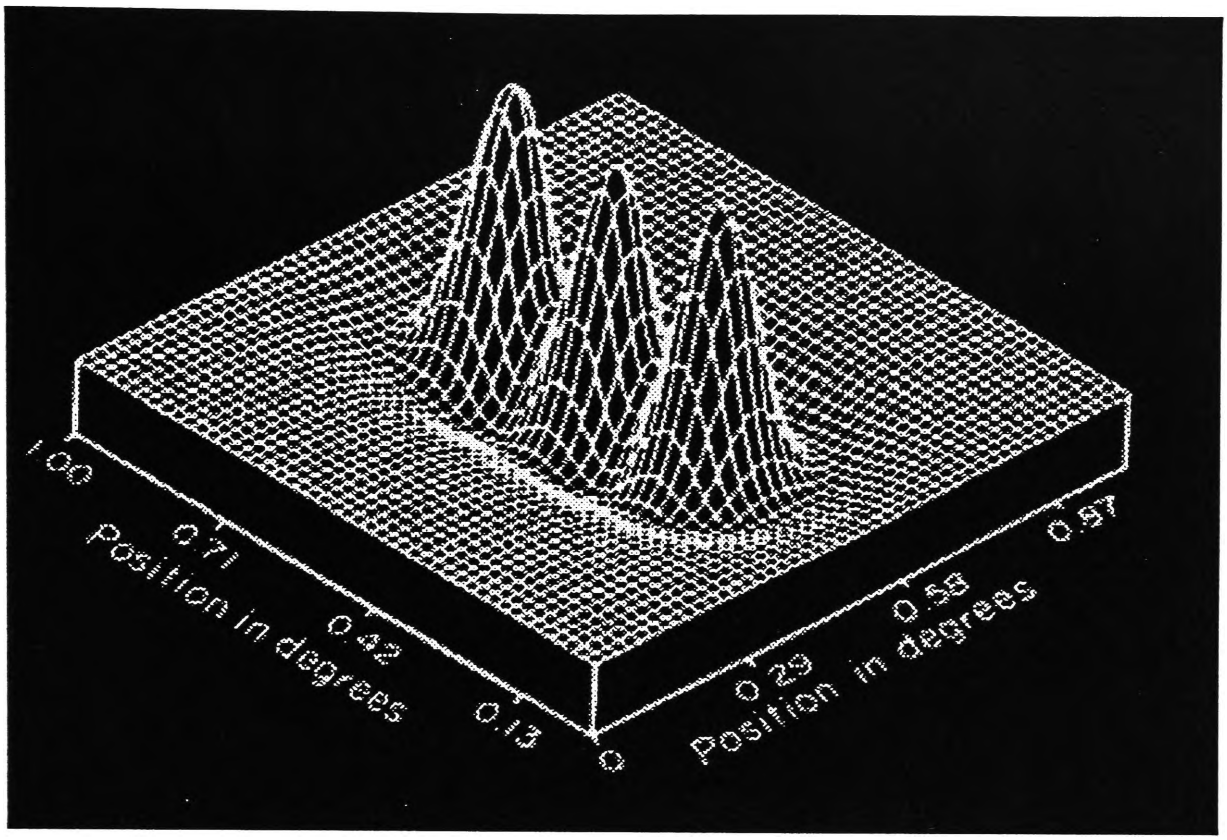

Figure 2.2: Three LGN cells concatenated

Difference Of Gaussian (DOG) i.e.

$$f(x) = e^{-x^2} - \frac{1}{6}e^{-x^2/9}$$

The DOG representation simulates a simple retinal cell with the assumption that one cell responds to only one change in light intensity as shown in figure 2.1. This assumption is incorrect since the simple cell responds to more than one change in light intensity. To cater to multiple changes in light intensity per retinal cell, we have to **concatenate multiple LGN cells** having DOG weighting functions characterizing each LGN as shown in figure 2.2 and figure 2.3. Figure 2.2 displays the concatenation of the response of three LGN cells having DOG weighting functions, and figure 2.3 shows the concatenation of nine LGN cells over the same geometric spread.

Here it should be noted that **the human visual cortex takes into cognizance the same number of changes per cell whatever the frequency** i.e. if the frequency is high the same number of changes in light intensity variation are compressed into a smaller width and for a low frequency the same number of changes in light intensity are spread out over a larger width as shown in figure 5.5 i.e. the cells react to different sizes.

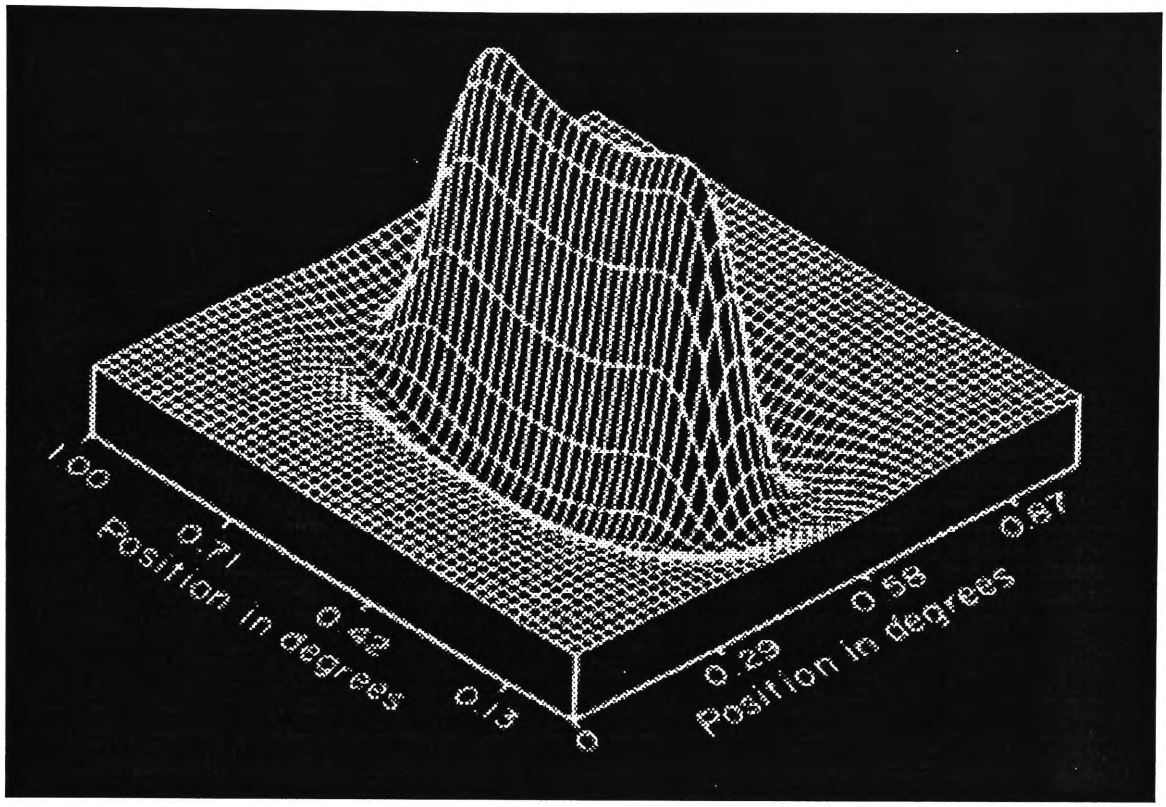

Figure 2.3: Nine LGN cells concatenated

In terms of machine vision this property of the human visual system of responding to different sizes translates to *spatial frequency* which is the number of changes in light intensity. This spatial intensity has to be localized to a neighbourhood in order to simulate the functioning of a simple cortical cell.

2.1.3 High Level Vision

The second and third of the three fields above is concerned with the interpretation of the information generated by the eyes - which involves the brain - that actually perceives the image and makes decisions as per the task at hand. This stage may be termed as high level vision. High level vision is principally concerned with perception and little information if any is available on its functioning.

In this project primary concern is the simulation of some of the low-level features of the human visual system.

2.2 Machine vision

Machine vision may be looked upon as an attempt directed towards emulating the human visual system. It is interesting to note that the eye has hundreds of data paths

converting raw image data into information in terms of intensity, phase, orientation and spatial location in the image space. On the other hand the video cameras that are mostly used in Robotic vision systems have a single data path. The camera image formation mechanism basically scans the image from left to right serially, generating a train of impulses proportional in amplitude to the light intensity either in interlaced or non interlaced mode. Video cameras that use arrays of CCDs are limited by the small size of the CCD array that is commercially available [Bro86].
Acquiring an image from a given scene is done by visual sensors like video cameras. These sensors convert visual information to electrical signals by sampling the scene at regular intervals in space. These electrical signals are quantized. This collection of electrical impulses provide a digital image i.e., a discrete image in a matrix form that is easily processed by machines.

2.2.1 Prerequisite for Machine Vision

An image is basically a two dimensional entity. It is a continuous distribution of light energy over a two dimensional surface (area).
The primary requirement for processing this image by either man (Human visual system) or machine is converting it into a sampled form, even though the sampled form of the image in case of the human vision system is somewhat different from that of the machine vision system.
For an image function $f(x, y)$ to be in a suitable form for machine processing, it must be digitized both spatially and in amplitude [CGW87]. Digitization in the spatial coordinates is called image sampling and digitization of amplitude is termed *gray-level quantization.*
A continuous image on being sampled into N rows and M columns and on quantization may be represented as an $M \times N$ matrix as follows, where M and N vary from zero to $(M-1)$ and $(N-1)$ respectively.

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \vdots & & & \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix}$$

Each element in this matrix is termed as a picture element which is also termed *pixel.* The sampling rate and intensity levels required to produce a useful (from the Machine vision point of view) reproduction of the original scene depends on the image itself and the final application for which the image is required.

2.2.2 Machine vision perspective of low-level vision

Since the hardware used in machine vision implementation does not work in the same fashion as the human visual system, The methodologies adopted for machine vision are different. The algorithms used too are not really a direct emulation of the human eye, but rather simulations of the processes in the human visual system.
Fixation in machine vision enables the system to reduce the amount of data to be processed, by concentrate on the relevant parts of image data leading to cognition and hence comprehension.
The input given to a machine vision system is the sampled distribution of a continuous function of light in two dimensions which means a digitized image. The output must satisfy certain conditions like

- (i) It must have some relationship with the input image
- (ii) It should provide the information necessary for the task for which the machine vision system was implemented.

Machine vision is closely related to the fields of

- (i) Image processing
- (ii) Pattern recognition
- (iii) Image perception or interpretation

In the human visual system, the first one of these fields is dependent on the mechanisms for image processing, built into the eye. Hence it may be termed *low level vision.*

Chapter 3

IMAGE FORMATION

"Common sense is inspite of,
and not the result of education".
– Victor Hugo –

An intelligent robot should be able to sense and react to its environment, and visual sensing is perhaps the most effective sensory mechanism.

The type of information required to make some sense of the surroundings is the brightness, depth, orientation, colour and texture of objects. The tools necessary to achieve this goal are either one or two cameras, an optical system, and the electronics necessary to communicate with the controlling computer [GN89].

The object under scrutiny together with its surroundings, is scanned, sampled, and input as a digital image. Other sensors may also input data about the object and surroundings. This is the first step in any machine vision system.

The data so collected is processed and redundant data is eliminated. Relevant data is collated and then used for the purpose of cognition, analysis and perception.

This chapter is about the process of capturing and forming of an image in a form that lends itself to computer processing. Before one can decide on the means of capturing an image, one has to understand the ultimate application for which the image is required. It would be instructive to have a brief look at the possible applications.

3.1 Components Of Machine Vision

To endow a robot with vision, it needs to have certain generalised components which constitute machine vision. A machine vision system must generate by some method a matrix of points which are light or dark depending on the scene under observation.

The factors involved in transferring a scene to a processing unit are:

- Illumination level on the object under observation.
- Conversion of the optical image into electrical information.
- Transferring the electrical information to the processing unit.

Conversion of the optical image involves the use of an optoelectronic converter that scans the brightness or the reflectance of the individual points in a given scene.

The factors that affect the selection of the image acquisition system are

- The number of picture elements into which the image is to be divided.
- The time available to acquire and process the picture.
- The environment in which the machine vision system has to operate.

Scanning systems for the purpose of image acquisition are by and large divided into two groups - viz. those that can scan a light spot across the scene and detect the strength of the returned light signal, and those within which an image scene scanned by an electron beam is formed inside the optoelectronic converter.

The second type finds wide applications and usage as will be seen [Bro86].

There are many different types of cameras available commercially - but the basic two technologies used are

1. The vidicon tube type
2. The charge coupled device (CCD) type

The optical system requirements are application and environment specific. For example the requirements could be as follows.

- Analog to digital conversion
- Initial processing, compressing etc. of the data

At times, if the camera is of the CCD type, the image data generated is of the digital type which may be processed using hardwired electronics for noise filtering etc. [Bro86] [Phi89].

3.2 Sensors

The different types of sensors used in the electronic generation of images are *Vidicon* and *CCD*.

3.2.1 Vidicon

A vidicon is basically the exact opposite process to that of a standard television using a cathode ray tube.

The working principle of a vidicon camera is exactly opposite to that of a TV CRT.

3.2.1.1 Image Formation

The image of the field of view is focussed onto a two dimensional plane which is made up of photosensitive material that generates electrical charge that finally gets converted to the electrical representation of the scene being scanned.

In tubes having an electron image building mechanism, the photosensitive phosphor emits electrons when the light incident upon it has sufficient energy (exceeding the critical value of the photosensitive layer).

Camera tubes that do not have an electron image formation mechanism, have a photosensitive layer that is of the photoconduction type. The scanning electron beam incident upon the photoconductive surface brings the potential on the surface near the potential of the gun. A positive potential is applied on the other side. The effect of incident light is to raise the energy level of the electrons so that the material starts conducting.

The increase in the energy level or potential due to the light that is incident, is directly proportional to the amount of light energy received by that element of the photoconducting surface until the next scan of the electron beam. The energy in the element is summated over time, i.e. charge is integrated with respect to time [KPH86] [Bro86].

The charge that is finally sensed by the electron beam at any given instant constitutes the video signal and represents the brightness of that point in the image.

3.2.1.2 Properties/Drawbacks

Due to changes in temperature and other factors the electrical parameters involved in the operation of the vidicon tube tend to drift and need adjustment from time to time.

The photosensitive layer is a flat plane that is scanned by the electron beam. As a result the beam is incident on the layer at an angle instead of being perpendicular to it. This can bring about distortion in the image being scanned due to the time difference involved in the beam scanning different parts of the flat layer.

The thickness of the layer affects the resolution of the tube. Even though the vidicon tube is cheaper, its negative aspects outweigh the cost factor. It is fragile, has a short life span due to the excessive heating of the electron emitting gun. An additional drawback is the low reliability in terms of *"mean time between failures"* and the higher rating of the power supply it requires as compared to the CCD technology. All these factors make it highly unattractive in the long run [CS89] [GN89].

3.2.2 CCD - Charge coupled devices

CCDs are basically made up of small elements (charge coupled devices) that are capacitive in nature and build up a charge over a period of time proportional to the intensity of light incident upon them.

Any number of these CCD elements can be put together in any shape to form a light sensitive surface of that shape. Most of the time these devices are used to form matrix structures of the order of $128X128$ or $256X256$ elements.

Each CCD cell consists of a semiconductor photosensor which generates charge pro-

portional to the intensity of light incident upon it. This photosensor is connected to a capacitive element which is also made of semiconducting material (thin or thick film) so that it can be embedded conveniently inside an IC.

A high stability clock circuit gates out the charge from the capacitive element at regular intervals for further processing.

The problem with this mechanism is of shifting the charge out of the capacitive element into a shift register, 100 per cent efficiency is not achieved. Some charge remains behind, and this can affect the final image quality. Then again the shifting out of a charge takes some time and this time can cut into the integration time of the photosensor. Despite all these factors which are usually countered by providing uniform bias diodes and parallel shifting of charge, and some mechanism for calibrating the output either in hardware or software gives a stable performance.

Considering factors such as having no moving parts, low heat dissipation, low rating of power supplies, high reliability due to mechanical ruggedness, etc. makes this an attractive and viable technology for cameras.

Resolution of the image obtained in a CCD camera is directly dependent on the size of each element. The smaller the element size the higher the resolution of the camera. Hence, very high resolution is possible. The CCD array is an integrated chip that can be mounted on a rugged platform. The whole set up (*the camera*) can be used even in highly hazardous conditions with scant effects on its performance. [Bro86].

3.2.3 Frame Buffers and Frame Grabbers

Interfaces between the sensor and the processing unit are normally assigned the task of digitizing the image obtained. If the image obtained is in analog form it has to be digitized before storage. An already digitized image, as in the case of commercially available CCD based circuits, only requires storage. Electronic circuit boards that combine the features of digitizing and storage are called *Frame Grabbers.*

In Frame Grabber systems, the image storage is in the form of a bit mapped structure. Each pixel in the image is represented by an $n - bit$ word indicating the light intensity level at that point in the image. The size of the image stored is set by the $n \times m$ pixel matrix. As the resolution of the image increases by a factor of 2, the buffer size required to store that image increases by 4.

Some Frame Grabbers also perform some additional functions like:

- Frame addition for averaging over several frames.
- Frame subtraction and remapping of pixels.
- Transforming the gray levels of selected pixels without disturbing other pixels in the image frame, by using a mask.

The additional functionality refers to the following:

- Frame addition can correct for low light levels and improve the *Signal/Noise* ratio in an image.
- Subtraction can eliminate the background data in an image to reduce the volume of data to be processed.
- Remapping of pixels helps in enhancement/segmentation.

3.3 Development System

The image source is a solid state camera that connects to the video input of a frame grabber transputer module. This module is part of the transputer network hosted by an IBM[1] compatible Personal Computer. Overall the system consists of the transputer network, with the root transputer connected to the IBM compatible PC/AT, and the camera connected at the video input of the frame grabber. The video out connection of the frame grabber goes to a monitor. So the captured image may be scanned before saving.

3.3.1 The Camera

The actual camera used for this project was a module from PHILIPS. This module has a built in solid state sensor, for sensing images. It also has dedicated circuits that are surface mounted on the board. This allows the module to function as a standalone imaging system. The printed circuit board is provided with buffering on the side of the video signal output, the other side has a lens mount attached to it. This imaging system provides an image of 256 grey levels and variable exposure control to sharp focus

[1] Any reference to IBM PC and the range of IBM Personal Computers implies IBM PC compatibles and no infringement upon any of the IBM trademarks

fast moving objects. The scan mode of this module can be selected externally by means of a control signal. Control signals may be used to set the pre-distortion to counter the distortion due to the picture tube. The gain factor is set to provide external synchronization. Switches are provided to set up modes like interlaced/progressive scan modes, Gamma on/off pre-distortion, gain control modes (manual/auto) and so on. In most situations no external synchronization is necessary and the connection marked "input" is left unconnected. A control signal to vary the brightness of the image is also provided.

Thus one acquires a digitised version of the actual image from the above setup.

Chapter 4

IMAGE PROCESSING/ANALYSIS

" Always remember that
mind is the chief fortress,
if you let it go,
you lose the battle".
– C. Rajagopalachari –

Capturing a digitized image is the first step in the process of machine vision. A captured image needs to be analyzed in the light of the fact that every pixel when considered as a part of the group of pixels that surrounds it, displays properties besides the intensity value at that point in the image. So, it is essential that one studies groups of pixels that together display some characteristics like spatial frequency, texture and so on.

Image is defined as a two dimensional light-intensity function, termed as $f(x,y)$. Every point in the (x,y) space is a number representing the intensity of the image at that point.

Any function of energy cannot be zero. Light being energy, any function of light intensity such as image is always non-zero and it cannot have values that tend to infinity.

The images that one sees are dependent on the light reflected from our surround-

ings. This reflected light is a function of the light incident upon an object and the amount (percentage) of light that an object reflects [CGW87].

Therefore, if $f(x,y)$ is an image function, one can say that

$$0 < f(x,y) < \infty$$

and

$$f(x,y) = i(x,y) \times r(x,y)$$

where i is the light incident at point (x,y)

$$0 < i(x,y) < \infty$$

and

r is the percentage of light reflected from point (x,y)

$$0 < r(x,y) < 1$$

The intensity of light at a point (x,y) in an image is termed as the gray-level of the image at that point.

This gray level lies within the minimum and maximum values of gray-level. The only requirement on these two bounds of gray level is that the minima be positive and the maxima be finite.

It is standard practice to call the minimum as zero and maximum as L

where,

gray scale = 0 is black

and

gray scale = L is white.

4.1 Image Discretization

When an image is discretized in space, it is termed 'image sampling' and when it is discretized in amplitude, it is called 'gray level quantization'.

At this point it should be noted that a discretized or digitized image is an approximation at best of the continuous function of light energy, i.e., the image.

How close an approximation a digitized image is to real life depends upon the number of gray levels assigned and the sample space used.

Higher resolution in terms of sampling and grey level coding results in higher computing power requirements.

The question of how good an image is, depends entirely upon the application for which it is intended. For some applications images at a lower resolution are sufficient. For example medical imaging.

Various mathematical techniques are combined for the purpose of image processing.

4.2 Types Of Operations

Three types of operations are possible for the preprocessing and enhancement of digital images.

- Pixel Based Operations,
- Neighborhood Operations,
- Image Content Dependent Operations

4.2.1 Pixel Based Operations

Also known as point operations, pixel based operations mostly affect the grey scale pattern of an image. After a pixel based operation, each pixel in the output image corresponds to one in the input image having the same coordinates.

For an input image $f(x,y)$ a pixel based operation will produce an output $f'(x,y)$, where

$$f'(x,y) = h(\, f(x,y))$$

and the operation is completely defined by the function, which is in fact a one to one mapping from input image to output image. Such operations may be of linear, logarithmic, or exponential type.

4.2.2 Neighborhood/Spatial filter operations

Image enhancement using spatial filtering techniques requires the operation to be spread over a neighborhood of a certain size. The input image is convolved with a filter *point spread function (psf)* with the expectation that the output will be an improvement on the input as far as certain features or characteristics of the image are concerned.

In a discretized digital image, the convolution[1] process at an individual pixel (x, y) is defined as:

$$f * G = \sum_{i=-m}^{m} \sum_{j=-n}^{n} f(x+i, y+j)G(i,j)$$

where

$$\begin{aligned} 2m+1 &= \text{no. of columns of filter mask} \\ 2n+1 &= \text{no. of rows of filter mask} \\ f(x,y) &= \text{image} \\ G &= \text{filter function} \\ x &= \text{x-coordinate of pixel in image} \\ y &= \text{y-coordinate of pixel in image} \end{aligned}$$

The principal approach used in defining a neighborhood about (x, y) is to use a *square or rectangular sub-image area centered at* (x,y). The center of the sub-image is moved from pixel to pixel, say starting at the top left corner, and applying the operator at each location (x, y) to yield the value of $f*G$ at that location. The image pattern can be convolved – shift-multiply-sum operations point by point, with a filter function that is translated over the entire pattern [Nib86] [MH83].

Besides image enhancements, masks may be used for image restoration, object segmentation and computing the skeleton of a binary region.

---

[1]Convolution is denoted by $*$

4.2.3 Image content dependent operations

In order to decide what is image and what constitutes noise, before any actual filtering for enhancement can be carried out, a low level analysis of the image contents is necessary. In the past, attempts have been made to find methods of taking into consideration the non stationary properties of image data [MH83]. A general operator model that allows representation of the image features as a hierarchical structure in terms of different level primitives was suggested by Granlund and Knutsson [P:G82].

The above model is essentially a two stage process. First the image is convolved with several filters of differing feature extraction properties like line and edge extraction. In the second stage, a filter for enhancement is constructed from the magnitude and direction of the sensed feature e.g. edge.

4.3 Objectives In Image Processing

The above mathematical operations and/or techniques are used in various processes involved in Machine vision.

Some of the processes involved in machine vision, wherein the above mentioned techniques are used are:

1. Image enhancement,
2. Image recognition,
3. Image analysis for perception,
4. Image restoration,

   Image coding/transform coding,

   Image recovery from distorted signal.

4.3.1 Image Enhancement

The term enhancement of an image implies some kind of operation upon the digitized image to improve it in some respect. Some property of the image is enhanced in order to facilitate a better understanding of the image for human or machine, depending upon the requirement at hand. At the same time one should not lose sight of the fact that enhancement leads to loss of information. One cannot obtain more information from the enhanced/transformed image than from the original one. One can only highlight some particular feature of the image for better understanding.

Image enhancement operations result in:

1. highlighting certain subjective properties,
2. elimination of noise,
3. interpolation or extrapolation effects to suppress any kind of distortions.

The human visual system does a near perfect job of filtering an image, especially filtering that is dependent upon the content of the image. At times the human system can interpret the raw image better than the processed image, unless the information is hidden as in the case of geophysical survey images or, in the case of medical imaging. The processing highlights and sharpens edges without a change in contrast and enhances the features that assist in making sense out of a fuzzy input image. What constitutes noise and what is a genuinely acceptable image is a highly subjective decision and depends upon the application.

A digitized image will normally contain some amount of noise. In order to separate the noise from the useful features in the image, some knowledge about the processed image is necessary. Some degree of preprocessing under a given set of rules is necessary to acquire this knowledge about the features in the image.

4.3.2 Image Restoration

Image restoration deals with:

- Extracting and rebuilding the original image from the compressed and / or encoded form,
- Reconstructing the original image from a distorted dataset, or partially lost dataset, by way of interpolation techniques or other algorithms.

4.4 2D Digital Signal Processing and Systems Theory

Over the years, one dimensional signal processing techniques have been generalized to two dimensional signals. This has resulted in highly efficient and fast algorithms for the purpose of image coding, image enhancement and image restoration related problems. Some of these techniques are also used for edge detection and feature extraction.

Multiferous techniques are available for the purpose of processing images in two dimensions. This research project concentrates on those techniques that are linear space or shift invariant (LSI).

Although the total process of image capture and processing is non-linear at the global level, at times even space/shift variant. It is to be noted that from the total process at least one step, such as the algorithm used (which may be a transformation of the image), is likely to be a two dimensional LSI filtering operation. Hence, one can have a closer look at LSI filters [KPH86].

When one says filtering an image, the underlying idea is to transform the image in some way and generate a new image that is easier to manipulate. Most of the methods used in this processing of images are linear shift invariant.

With the help of linear and shift-invariant system theory, one can discuss concepts like convolution, use of spatial frequency and the transformation from spatial domain to frequency domain, transform coding of images for compression and/or filtering, and study the transform domain at large.

4.5 LSI Systems

A two dimensional system, represented by the operation h, that produces for an input of $f_1(x,y)$ an output of $g_1(x,y)$, and for input $f_2(x,y)$ an output of $g_2(x,y)$ is called linear if

$$h(\alpha\, f_1(x,y) + \beta\, f_2(x,y)) = \alpha\, g_1(x,y) + \beta\, g_2(x,y)$$

and shift invariant if

$$h(f_1(x - x_0, y - y_0)) = g_1(x - x_0, y - y_0)$$

It can be proved that a system whose response can be described by a convolution, is linear and shift invariant and vice versa [KPH86] and [Nib86].

4.5.1 Linear Operations

Representing the image as a matrix of light intensity values allows one to manipulate the image by using linear Algebra. One may redefine the vectors of an image as the linear combination of the original vectors [HJ90].

For example one can define a 4 vector image transformation as:

$$\begin{pmatrix} outputvector1 \\ outputvector2 \\ outputvector3 \\ outputvector4 \end{pmatrix} = \begin{pmatrix} A & B & C & D \\ E & F & G & H \\ I & J & K & L \\ M & N & O & P \end{pmatrix} \times \begin{pmatrix} inputvector1 \\ inputvector2 \\ inputvector3 \\ inputvector4 \end{pmatrix}$$

In general a linear operation with the above representation can be explained as:

$$\begin{aligned}
outputvector1 &= A \times inputvector1 + B \times inputvector2 + \\
&\quad C \times inputvector3 + D \times inputvector4 \\
outputvector2 &= E \times inputvector1 + F \times inputvector2 + \\
&\quad G \times inputvector3 + H \times inputvector4 \\
outputvector3 &= I \times inputvector1 + J \times inputvector2 + \\
&\quad K \times inputvector3 + L \times inputvector4 \\
outputvector4 &= M \times inputvector1 + N \times inputvector2 + \\
&\quad O \times inputvector3 + P \times inputvector4
\end{aligned}$$

These linear algebraic operations allow a very flexible manipulation of the image data space. In algebraic terms the operation is just a linear combination of the pixel values in a image vector. But in geometric terms a large variety of sophisticated transformations can be effected by these operations.

4.5.2 Orthogonality

In mathematical terms the image in matrix form allows one to represent the changes in spectral properties of an image in algebraic notation. The concept of Orthogonality has a bearing on what one can expect to see at the end of an image processing operation.

Any image transformation that is *orthogonal* is reversible i.e. one can recover the original image from the transformed image by a straightforward inversion operation. But in case of those image transformations that are not orthogonal, the recovery of the original function is not that easy. The concept of orthogonality of a transform is therefore important.

4.5.2.1 Transformation and Orthogonality in One Dimension

The property of orthogonality is fundamental to representation of a data vector in terms of a set of predetermined basis vectors and the associated transform coefficients.

The set of basis vectors allows one some latitude in approximating the desired function f with a limited number of basis vectors and their associated weighting coefficients C_p. The addition of further terms to reduce the error involved in the approximation do not mean modifications to the previously determined values of C_p, if the transformation is orthogonal. If the transformation is orthogonal, the cross product terms that appear are automatically zero. In the case of the Fourier transform, the product integrals that result from the expansion are zero.

$$\int_0^T sin\ nu_0 t.dt = 0$$

$$\int_0^T cos\ nu_0 t.dt = 0$$

and

$$\int_0^T sin\ ku_0 t.cos\ nu_0 t dt = 0$$

$$\int_0^T sin\ ku_0 t.sin\ nu_0 t dt = 0\ k \neq n$$

$$\int_0^T cos\ ku_0 t.cos\ nu_0 t dt = 0\ k \neq n$$

One can also say that if the set of basis vectors is complete (with a finite number of terms it is always possible to represent a continuous signal with an approximation which differs from the original by an amount less than ϵ, however small the value of ϵ may be), then one can represent the data vector exactly by a finite number of basis vectors and weighting factors [JC85].

Transform relationships can be expressed concisely in matrix form as follows:

$$\hat{f}(u) = [Z]f(x)$$

In order to recover the original signal one needs to be able to invert the set of basis vectors given above i.e. Z. Multiplying both sides of this equation by the inverse of the transform matrix will give:

$$[Z]^{-1}\hat{f}(u) = [Z]^{-1}[Z]f(x)$$

$$[Z]^{-1}\hat{f}(u) = [I_N]f(x)$$

Where $[I_N]$ is the unit or the identity matrix of order N.

Hence

$$f(x) = [Z]^{-1}\hat{f}(u)$$

In case of orthogonal and unitary matrices, finding the inverse is simple. In case of an orthonormal unitary matrix, having complex terms, the relationship may be

depicted as follows

$$[Z_C]^{-1} = [Z_C^*]^T$$

The basis matrix inverse is the transpose of its complex conjugate. The subscript C denotes that the matrix is complex. In general when the unitary matrix is not orthonormal, the orthogonality for real variables may be stated as:

$$[Z_R]^{-1} = \frac{1}{N}[Z_R^*]^T$$

Where $N = 1$ for an orthonormal basis matrix. The significance of the above statement is that, the original data may be recovered, without the need for matrix inversion. For example the Fourier transform [JS89].

In the case of those transforms that are not orthogonal, retrieval of the original function from the transformed data is possible, even if somewhat complicated. An auxiliary function has to be worked out that makes the transformation orthogonal and the inversion possible.

The above results may be generalized to the two dimensional form, in case of the class of transformations that belong to the $L^2(R)$ space.

4.5.3 Impulse Response and Convolution

The result from linear systems theory is that if a filter satisfies the conditions of linearity and shift invariance in the discrete one dimensional case the output of the filter can be expressed as [Nib86]:

$$g(x_i) = \sum_{k=-\infty}^{\infty} f(x_k)\, h\,(x_i - x_k)$$

Where $h(k)$ is called the impulse response of the filter and completely characterizes the filter.

The generalized expression for the above, in two dimensions may be expressed as:

$$g(x_i, y_j) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f(x_k, y_l)\, h\,(x_i - x_k, y_j - y_l)$$

In the two dimensional case, if one defines a filter mask of size $[i-w, i+w] \times [j-v, j+v]$, the same may be expressed as:

$$g(x_i, y_j) = \sum_{k=i-w}^{i+w} \sum_{l=j-v}^{j+v} f(x_k, y_l)\, h\,(x_i - x_k, y_j - y_l)$$

The above operation is basically a convolution operation in the spatial domain (defined in the section on spatial filtering) and,

$$h(x_i, y_j)$$

is termed the *point spread function*, i.e. the impulse response in two dimensions. If $f(x, y)$ is a bright point at the origin of the input image then

$$h(x_i, y_j)$$

indicates how the system spreads out that point of light. That is why it is called the point spread function. The Fourier Transform of this function is also known as the *Modulation transfer function* [KPH86] [Nib86].

The point spread function of a filter characterizes the filter completely and the values of h are termed as filter weights, filter kernel, or the filter mask.

4.5.4 Separability

At times $h(x, y)$ is represented as a product of two components h_x and h_y, one in the vertical direction and the other in the horizontal i.e.

$$h(x_l, y_m) = h_{vert}\,(x_l) \times h_{horiz}(y_m)^T$$

If the above condition is satisfied, and if the filtering operation can be carried out separately in the vertical and horizontal direction, then the filter point spread function $h(x, y)$ is said to be separable. This property is very useful since the image may be convolved in one dimension, and then the output of the first convolution can be convolved again, for filtering in another dimension. This reduces a single but complicated two dimensional filtering operation, to, two simple one dimensional filtering operations.

In optical systems the point spread functions are always non negative. Some physical as well as biological systems can have negative point spread functions. For example the DOG functions of the retinal ganglion cells. For the purpose of digital image processing, these values may be altered arbitrarily, giving rise to filters that are non-linear, adaptive and of varying window sizes. These functions may be termed as moving win-

dow type, or boxcar type filters that are no longer linear. This flexibility in defining the point spread function maybe used to define filters for various applications like feature extraction or smoothing, for removing noise, or for data reduction by segmentation for the purpose of cognition etc.

4.6 Segmentation Techniques

There is no complete theory developed for the purpose of image segmentation, but over the years a number of techniques have been developed. Segmentation is one of the most important steps in automated image analysis either for image extraction or cognition. Segmentation is be achieved by either detecting the edges/contours of objects in an image, or segregating texture based regions. Both methods will be discussed in this section. Segmentation by way of edge detection is mostly used when it is known that the object under observation does not have any feature of interest that is fine grained in texture.

The techniques used for the purpose of segmentation principally rely on two approaches relating to the grey-level values of the pixels. These are:

- discontinuity or sudden change in grey-level i.e. edge detection.
- Similarity in grey-level values of pixels i.e. textural segregation.

The techniques based on the discontinuity of grey-level values, normally use small spatial masks. These masks are convolved for filtering purposes as detailed in the section on spatial filtering.

Edge detection may be useful as a segmentation technique in situations where the approximate shape of the object to be segmented is known, and does not have fine grained texture which may be construed as an edge.

4.6.1 Edge detection

In most edge detection techniques that are commonly used, the underlying idea is of a localized derivative operator.

The first derivative at any point in any image can be obtained by computing the magnitude of the gradient at that point.

4.6.1.1 Gradient

The Gradient G, at any point (x, y) in an image $f(x, y)$ is defined as follows:

$$G[f(x,y)] = \begin{bmatrix} \frac{df}{dx} \\ \frac{df}{dy} \end{bmatrix}$$

The vector $G[f(x,y)]$ points in the direction of maximum rate of increase of the image data function $f(x,y)$ and its magnitude is equal to the maximum rate of increase of $f(x,y)$ per unit sample space.

The gradient magnitude is given by

$$|G[f(x,y)]| = [G_x^2 + G_y^2]^{\frac{1}{2}}$$

This magnitude is in the direction of vector G, where the direction of vector G is given by

$$\alpha(x,y) = tan^{-1}[(\frac{df}{dx})/(\frac{df}{dy})]$$

4.6.1.2 Laplacian

The Laplacian is defined as:

$$G^2[f(x,y)] = \frac{d^2f}{dx^2} + \frac{d^2f}{dy^2}$$

The approximation of this function used is given by:

$$G^2[f(x,y)] = f(x+1,y) + f(x-1,y) + f(x,y-1) + f(x,y+1) - 4f(x,y)$$

4.6.1.3 Problems with Derivative based operators

The problem with these derivative based operators is that they sense the change in pixel value which makes these operators very noise sensitive. The average image considered will have sufficient noise that the Gradient and/or the Laplacian will consider it as an edge.

4.6.2 Texture Based Segmentation

Texture is one of the important characteristics used in identifying objects or regions of interest in an image. In a search for meaningful features for describing pictorial information, it is only natural to look toward the type of features which human beings

use in interpreting pictorial information. Spectral, textural, and contextual features are three fundamental pattern elements used in human interpretation of colour photographs.

Spectral features describe the average tonal variations in various bands of the visible portion of the electromagnetic spectrum, whereas textural features contain information about the spatial distribution of tonal variations within a band. Contextual features contain information derived from blocks of pictorial data surrounding the area being analyzed. When small image areas from black and white photographs are independently processed by a machine, then texture and tone are most important.

The concept of tone is based on the varying shades of gray of resolution cells in a photographic image. Texture is concerned with the spatial (statistical) distribution of gray tones. Context, texture, and tone are always present in an image, although at times one property can dominate the other. Texture is an image property of virtually all surfaces. It contains important information about the structural arrangement of surfaces and their relationship to the surrounding environment [JHSD73].

Some of the problems encountered within the context of texture analysis could be that of classification and discrimination, description, and segmentation. These problems are listed in order of increasing difficulty and it is clear that the major problem is that of texture segmentation. Segmentation is the partitioning of an image into regions that are *homogeneous* with respect to one or more characteristic. One of the basic issues to be considered is that of the cell unit size, i.e. the resolution of the area over which measurements are made to test for homogeneity.

4.6.2.1 Grain size and spatial resolution

Resolution in space is related to the size of the smallest detail that can be detected. Just as one needs to see the smallest detail in space, we also need to resolve the regions having differing textures. Different textures translate to differing spatial frequencies. The need for high resolution in both the spatial and frequency domains leads to representations, wherein it is known that one cannot simultaneously achieve arbitrarily high resolution in both domains. Research workers in both fields of Human Visual System and Computer Vision have become aware of this limitation [RRW90] and are concentrating on the conjoint domain.

There is evidence that visual discrimination is a local process [RRW90]. The rel-

atively poor performance exhibited by early frequency analysis methods, which were global in nature, can thus be explained. Visual effects that could not be explained using global frequency analysis methods might also be accounted for if local analysis is used.

As one must have observed from the above discussion, segmenting out textures with regard to their localized variations requires a localized approach. This should not be computation intensive or else it would be unattractive for real-time applications.

4.6.3 Methods in texture separation

Two principal methods used in Textural segmentation may be classified as:

- STATISTICAL METHODS : Statistical features of the image to be analyzed are selected with one primary criterion; that of minimal correlation between any two selected features.
- SPATIAL/SPATIAL - FREQUENCY METHODS : These methods are primarily based on image representations in the conjoint spatial-frequency domain. Such methods are able to achieve reasonably high resolution in both domains (*s/sf*) and they are consistent with recent theories in the field of Human Visual System [GD80], [GD85].

4.6.3.1 Statistical features for texture

Since the textural properties of images appear to carry useful information for discrimination purposes, it is important to develop features for texture.

To measure texture, the properties of the pixels are measured within a small window which is moved over the surface. There are two kinds of texture, regular, in which variations form a pattern which repeats cyclically over the surface, and statistical in which the variations never repeat exactly but maintain constant statistical properties. The properties of the texture in a gray scale image of a rough surface, conveys information regarding the mechanical profile of the surface. In other words the degree of roughness or unevenness may be gauged from the textural features.

Image texture studies have employed statistics like:

- Autocorrelation functions.

- Power spectra.
- Restricted first and second order Markov meshes.
- Relative frequency of various gray levels on the unnormalised image.

Recent attempts to extract textural features have been limited to developing algorithms for extracting specific image properties such as coarseness and presence of edges. Many such algorithms have been developed and tried on special imagery [MH83], [JHSD73].

4.6.3.2 Spatial/Spatial-Frequency Representation

The representations in the Spatial-Frequency that are widely used are:

- THE SPECTROGRAM: The spectrogram of an image is simply the squared magnitude of the Window Fourier transform of an image.

- THE DIFFERENCE OF GAUSSIANS (DOG) REPRESENTATION

  The DOG representation of an image is the sum of the outputs of a bank of 2D bandpass filters. This is similar to the model proposed by Wilson [WCG77] for the human visual system.

  Even though a lot of attention has been focussed on the multiresolution, pyramidal representation, which consists of a number of filtered outputs of the original image, the DOG has received special attention because of the correlation between the response of DOG and retinal cells. This is detailed in section 2.1.2. The DOG is a three dimensional representation. Besides the two image dimensions it also represents the center frequency of the filtered image for a given resolution.

  The impulse response of the filter at any predetermined resolution may be expressed as:

$$\begin{aligned} h(x,y) &= h_1(x,y) - h_2(x,y) \\ &= A\, e^{-a_1x^2-a_2y^2} - B\, e^{-b_1x^2-b_2y^2} \end{aligned}$$

  The DOG power representation may be formed from the squared outputs of these filters [RRW90].

James Crowley and Parker [LC82] have implemented a generalized version of DOG, i.e. DOLP, and Crowley has studied the SDOG (Sampled DOG) representation.

- WINDOW FOURIER TRANSFORM AND VARIANTS.

  The Window Fourier transform has the property of localizing the effect of the Fourier transform in space position. In effect this is like the visual discrimination feature in human vision and is a local process. This project uses this property of the Window Fourier transform for the simulation of the preattentive mechanism.

These techniques based on the Window Fourier transform will be discussed in detail in the next chapter.

These are a sampling of some of the most commonly used methods for segmentation. The principal consideration for the method of segmentation is the mode of image representation.

Chapter 5

SPATIAL/SPATIAL-FREQUENCY BASED SEGMENTATION

"Mathematics is the only science where,
one never knows what one is talking about,
nor whether what is said is true".
– Bertrand Russell *b.* 1872 –

5.1 Gabor model for receptive field profile

The ability of the human visual system to distinguish objects partially depends upon the perception of similar and dissimilar textures. Experiments in psychophysics suggest that certain texture types can be distinguished, or segmented by the preattentive mechanism of the human visual system. Features like colour, size, brightness, and edges can also be detected by the preattentive mechanism.

The results of computer modeling suggest that the simple cells in the human cortex have receptive fields that show a behaviour pattern very similar to Gabor functions. Thus, besides being a good operator for extracting textural information [Tur86], the 2D Gabor function serves as a good model for simple cell receptive field profiles.

Experimental results [Tur86] suggest that an algorithm for automatic segmentation of textures of interest might be constructed on the basis of Gabor functions. The type of computation involved might be suitable for certain kinds of parallel processing.

The physiological community through the 1970's was enamoured by two models used to describe the behaviour of the striate cortex. The two models were, "the bar type feature detector", and the "sinusoidal grating". It was only after the proposal of Gabor's model as a better fit for the performance of simple cell receptive fields, that research workers percieved that these two approaches were the extrema of a continuum. Greater resolution in one domain resulted in a decreased resolution in the other, [GD80], [GD85].

In particular, 2D Gabor filters have been found to be useful for analyzing textured images that display specific frequencies and orientation characteristics. The segmentation achieved by Clarke *et al* for natural as well as artificial textures, and by Turner for artificial textures in psychophysical experiments, bear a striking resemblance to human perception [Bea90].

Texture segmentation basically relies on the differences in the dominant characterizing frequencies in mutually distinct textures. By encoding images into multiple, narrow spatial frequency and orientation channels, textural regions can be segmented.

Multifrequency channel decompositions have been successful in explaining some low-level biological processes.

The Gabor expansion of a function into several frequency channels provides a representation that is intermediate to the Fourier and the spatial representation of a signal.

The filters used for these channels are 2D Gabor filter functions. These functions are useful for the purpose of segmentation due to various reasons, namely:

- Tunable orientation,
- Radial frequency bandwidths,
- Tunable center frequencies,
- Achieving optimal joint resolution in space and in spatial-frequency.

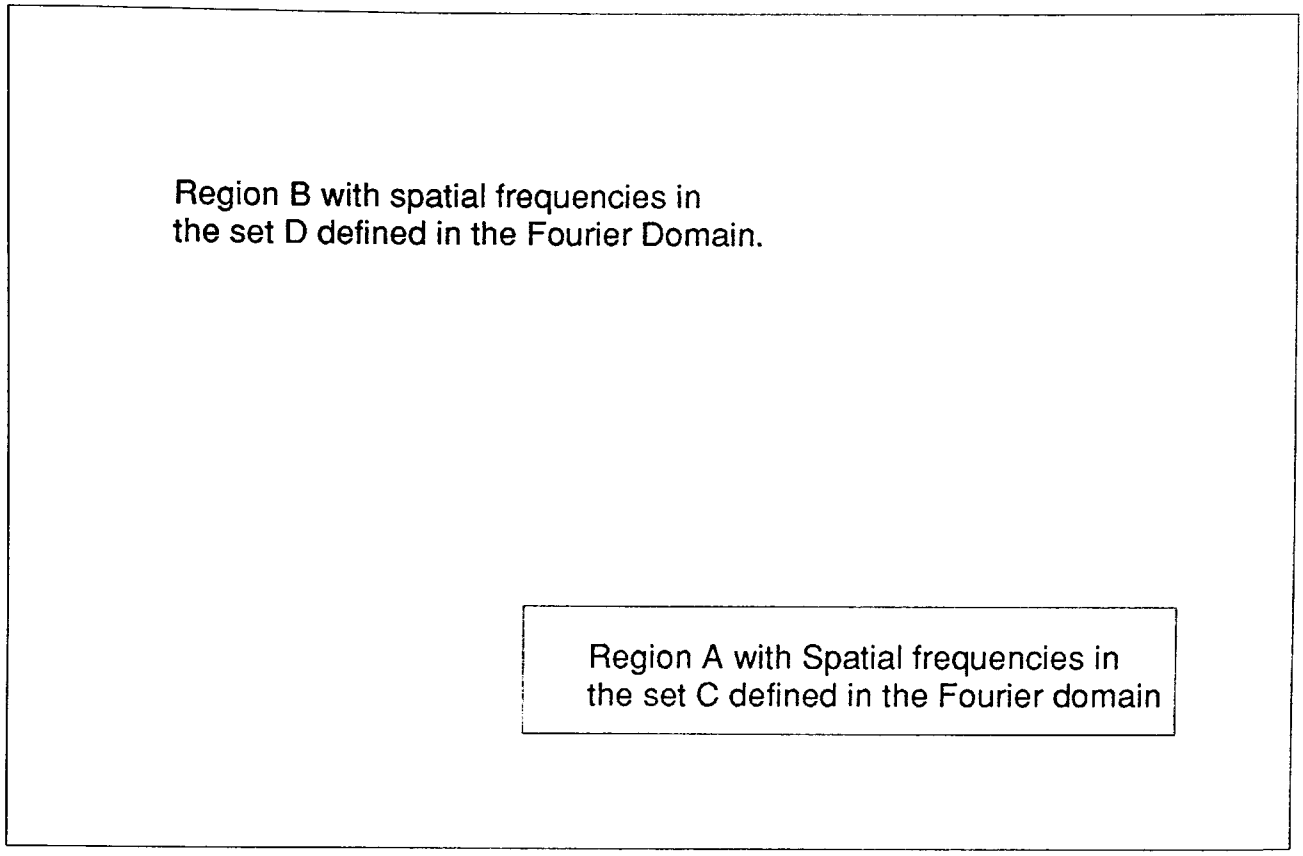

Figure 5.1: Idealized image model used for mathematical formulation of the problem

Image filtering in the spatial domain is often implemented as a convolution of the image data with a two dimensional mask suitably defined to carry out the filtering operation. For large mask sizes the convolution process requires immense number crunching ability on the part of the computing machine [Ran91].

5.2 Formulation of the problem

Consider an image of size 256×256. in which the region containing text is denoted by A and the rest of the image is denoted by B. Such an image is shown in Figure 5.1. Region A is the area of interest and the objective is to extract the text in the region. An image that can be modelled this way is shown in Figure 5.1.

The idealized image of Figure 5.1 can be used to formulate a mathematical statement of the problem.

The properties that can be used to segment out texture in an image are the spatial

Figure 5.2: The image representing mathematical model

frequency and the direction/orientation within the region containing the texture.

An image that represents the case of the idealized model is shown in Figure 5.2

The Mathematical statement of the problem will be as follows:

Let $f(x,y)$ and $g(x,y)$ be two functions in two dimensions defined over the interval $(-\infty,\infty)$. Let A and B be regions in this two dimensional space such that, $f(x,y)=0$ for all (x,y) in B and $g(x,y)=0$ for all (x,y) in A.

In equation form we can write:

- $f \in R^2 = (x,y) : -\infty < x,y < \infty$
  and if $A \in R^2$,
  then
  f is supported by A in spatial domain
  if $f(x,y)=0 \quad \forall\ (x,y) \in B.$

- $g \in R^2 = (x,y) : -\infty < x,y < \infty$
  and if $B \in R^2$,
  then
  g is supported by B if $g(x,y)=0 \quad \forall (x,y) \in A.$

f is supported by A
and g is supported by B
where $A \cap B \neq \phi$

$\hat{f}$ denotes the Fourier transform of the function f
$\hat{g}$ denotes the Fourier transform of the function g

Let $\hat{f}$ be supported by C (in the frequency domain)
and $\hat{g}$ be supported by D (in the frequency domain - frequencies other than in C)
where
$C \cap D = \phi$ in the Fourier domain (Ideal case).

In order to define a band pass filter such that it retains all frequencies in A and ignores the rest (B), it is required to define a filter function that has a frequency characteristic containing the frequencies in the region of interest. This requirement of filter definition may be stated as follows [VN92].

$G * (f + g) = f$
and
$\hat{G}(\hat{f} + \hat{g}) = \hat{f}$
$\hat{G}\hat{f} + \hat{G}\hat{g} = \hat{f}$
$\hat{G}\hat{f} - \hat{f} + \hat{G}\hat{g} = 0$
Therefore G has the property that
$\hat{G}\hat{f} = \hat{f}$ *and* $\hat{G}\hat{g} = 0$

i.e. the above may be restated in spatial domain as:

$$G * g = 0 \ and \ G * f = f$$

and

$$G * (f + g) = f$$

If
$\hat{G}_{x,y}(u, v) = 1 \quad \forall u, v \in C \quad and \ \hat{G}_{x,y}(u, v) = 0 \quad \forall u, v \in D$

then G satisfies this property,

i.e. $\hat{f}$ is supported by C *and* $\hat{g}$ is supported by D in the Fourier domain

where

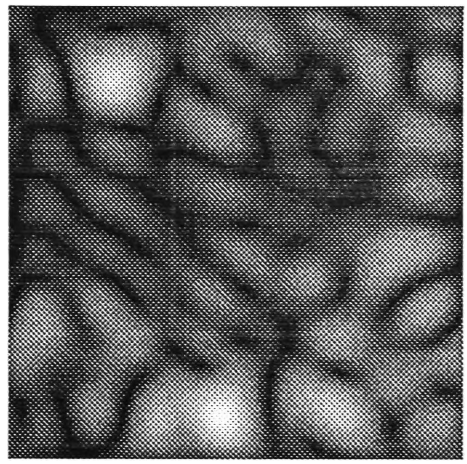

Figure 5.3: The bandpass filtered image.

$C \cap D = \phi$ (considering the ideal case)

This defines the bandpass filter for frequencies contained in region A.
There can be more than one solution to the problem of finding G, such that,
$\hat{G}(u,v) = 1 \ \forall u, \ v \ \in C \ and \ \hat{G}(u,v) = 0 \quad \forall (u,v) \ \in D$
One of the possible solutions could be
$\hat{G} = \chi_C$ (where χ_C is the boxcar window defined over C)
then $G = (\hat{G})^{\vee} = (\chi_C)^{\vee}$

If one tries to filter the image function with a bandpass filter, having a boxcar window defined over the dominant frequency of $(u_0 = 51, v_0 = 42)$, with a bandwidth of $(+4, -4)$, without thresholding, the results are as shown in Figure 5.3. This is certainly not what is desired.

Figure 5.3 on being thresholded at an intensity value of 125, gives us an output as in Figure 5.4.

The above approach in the frequency domain is unable to produce results because a band pass filter is a global operator. It has the drawback that the boxcar window in the frequency domain, on inverse Fourier transformation, acquires the shape of a sinc function in the spatial domain. This sinc function does not have the property of sensing a fixed size, i.e. a fixed number of variations in light intensity, at a specific frequency. This filter does not have the feature of detecting a particular orientation, and nor does an LGN cell simulated by a DOG representation detailed in Section 2.1.2 and Section 5.5. The results displayed in Figure 5.3 and Figure 5.4, suggest that the band pass filter detects the frequencies within its bandwidth at a global level, without any consideration to orientation and spatial-frequency of the texture.

In effect, one cannot rely on the frequency domain representation of the image to

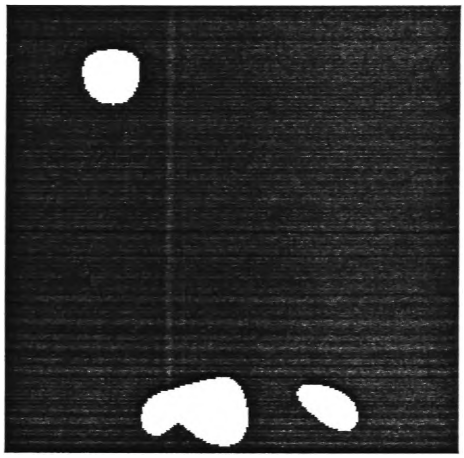

Figure 5.4: The bandpass filtered image after being thresholded.

get the desired simulation of the simple retinal cell.

The spatial frequency contained in the region of interest is local to its neighborhood. One is essentially looking for a localized operator that will do the filtering of the image, such that only the region having a particular texture is segmented.

The above problem suggests that there is a need for something like the Window Fourier transform (in two dimensions), or the short-time Fourier transform (STFT in one dimension).

5.3 Window or short-time Fourier transform

Any Spatial-Frequency representation will in essence be a *variant or a special case* of the Window Fourier transform, or what is known as the Short-time Fourier transform in one dimension. Studying the behaviour of the Window Fourier transform gives a better understanding of other Spatial-Frequency representations.

The aim of most signal analysis techniques is to extract relevant information from a signal by transforming it to another representation. The relevant information becomes apparent and is easier to use. Such transformations have long been applied to statistically stationary signals. For such signals the stationary transform is the well known Fourier Transform

$$\hat{f}(u) = \int_{-\infty}^{\infty} e^{-ixu} f(x) dx$$

The value of $\hat{f}$, defines the idea of global frequency in a signal. Fourier analysis

works well if the signal is composed of a few stationary components (e.g. *Sinewaves*). Any abrupt change in space in a non-stationary signal is reflected all over the frequency axis in $\hat{f}$. Therefore, the analysis of non-stationary signals calls for more than the Fourier transform.

Usually the approach adopted is to introduce space dependency in the Fourier analysis and preserve linearity at the same time. The principal idea is to introduce a parameter local in space—the "local Fourier transform" looks at the signal through a window over which the signal is approximately stationary [GM89], [RV91], [VN92].

In order to get the best of both worlds, we need a representation that is essentially a localized operator rather than a global operator like the Fourier transform. This also takes into consideration the spectral/textural properties of the image. One such signal representation in one dimension was proposed by Gabor in 1946, in his presentation on *"Theory Of Communications"* [Gab46]. This project describes an algorithm to distinguish between different textures using Gabor representation.

In 1946, Gabor in his paper *"Theory of Communications"* put forward the argument that the concept of frequency as interpreted through the Fourier Transform cannot explain certain physical phenomenon that insist on a description in terms of both space and frequency.

Signals can be represented in two dimensions with space and frequency as coordinates. Frequency of a signal that is not infinite in spatial extent, can be defined only with a certain inaccuracy. This is inversely proportional to the spatial extent [Gab46]. This uncertainty relation, akin to the one in quantum physics, suggests a new method for signal representation. It is intermediate between the two extrema of spatial analysis and spectral analysis. Any signal can be expanded in terms of elementary signals, by a process that includes spatial domain analysis and frequency (Fourier) domain analysis as extreme cases [Gab46].

In other words Fourier transform of a function $\hat{f}(x)$, gives a measure of the irregularities in a signal, but it does not pinpoint the location where the irregularities occur, i.e. this information is not spatially localized.

The Fourier transform is in fact defined as an integral over the entire spatial domain. As a result it is difficult to find exactly where the irregularities occur.

By defining a window $g(x)$ in the spatial domain within the Fourier integral, Gabor [Gab46] suggested a method to localize the information provided by the Fourier transform. This window is translated along the spatial axis to cover the entire signal.

The generalized definition of this Window Fourier transform or the short-time Fourier transform, at a position $x = x_0$, and for a frequency u, of a function $f(x)$

is given by,

$$Gf_{x=x_0}(u) = \int_{-\infty}^{\infty} e^{-\imath ux}.g(x - x_0)f(x)dx$$

The window function $g(x)$ at x_0 may take on any form, and the *integral* is now called a Window Fourier transform [GM89].

For the purpose of normalization it is considered that the energy of g(x) is equal to 1.

i.e.

$$||g||^2 = \int_{-\infty}^{\infty} |g(x)|^2 dx = 1$$

To analyze the spatial or frequency representation of the Window Fourier transform, consider σ_x the standard deviation of the window function $g(x)$.

$$\sigma_x^2 = \int_{-\infty}^{\infty} x^2 |g(x)|^2.dx$$

and σ_u the standard deviation of the Fourier transform of the window function $g(x)$.

$$\sigma_u^2 = \int_{-\infty}^{\infty} u^2 |\hat{g}(u)|^2.du$$

The function $g_{u_0,x_0}(x)$ is centered at frequency u_0 and has standard deviation σ_x in the spatial domain.

Now if the function is defined as:

$$g_{u_0,x_0}(x) = e^{\imath u_0 x}.g(x - x_0)$$

The Fourier transform of $g_{u_0,x_0}(x)$ is given by

$$\hat{g}_{u_0,x_0}(u) = \int e^{\imath u_0 x}.\hat{g}(u - u_0)e^{-\imath x_0 u}$$

This function is centered at u_0 and has a standard deviation σ_u.
Applying Parseval's theorem to the window Fourier transform defined above, we get the following.

$$\begin{aligned} Gf_{x=x_0}(u_0) &= \int_{-\infty}^{\infty} f(x)\overline{g_{u_0,x_0}(x)}.dx \\ Gf_{x=x_0}(u_0) &= \int_{-\infty}^{\infty} f(u)\overline{g_{u_0,x_0}(u)}.du \end{aligned}$$

In the above equation the first integral means that the Window transform depends

on the values of $f(x)$, $\forall\, x \in [x_0 - \sigma_x, x_0 + \sigma_x]$. The second integral means that in the frequency domain the values of Window transform $Gf_{x=x_0}(u_0)$ depend upon the values of $\sigma_u]$. Thus, the conjoint spatial/spatial-frequency (*s/sf*) domain in which $Gf_{x=x_0}(u_0)$ is defined may be represented by the resolution cell $[x_0 - \sigma_x, x_0 + \sigma_x] \times [u_0 - \sigma_u, u_0 + \sigma_u]$.

If $g(x)$ is gaussian, the Window Fourier Transform becomes the Gabor Transform.

5.3.1 Gabor Transform

The Gabor transform is a special case of a Window Fourier transform where the window $g(x)$ is gaussian.

It is well known that the 1D Gaussian function,

$g(x) \propto e^{-x^2/2\sigma^2}$

which has the Fourier representation of

$\hat{g}(u) \propto e^{-2.\pi^2.\sigma^2.u^2}$

is the only function mapping $R \Longrightarrow R$ that achieves the lower bound of the uncertainty relationship $\Delta x.\Delta u \geq \frac{1}{4\pi}$ relating the variances of $g(x)$ and $\hat{g}(u)$ respectively. In 1946 Gabor [Gab46] proved that this uncertainty relationship holds good for the complex valued 1D Gabor functions, and are the only class of 1D functions mapping $R^2 \Longrightarrow C$ to achieve the lower bound. Daugman [GD85] extended the result to two dimensions, and proved that the 2D Gabor functions achieved the lower bounds of the inequalities [Bea90].

$\Delta x.\Delta u \geq \frac{1}{4\pi}$

and

$\Delta y.\Delta v \geq \frac{1}{4\pi}$

Thus, the 2D Gabor filters achieve optimal resolution and localization simultaneously in spatial and spatial frequency.

The one model that shows properties similar to that of the human cortical cells is the Gabor filter in two dimensions—mimics the behaviour pattern of the cortical cells as far as spatial localization, spatial frequency selectivity, and orientation are concerned [GD88].

Gabor filters in the spatial domain are rectangular masks of dimensions $m \times n$ where often $m = n$ for reasons of symmetry. These localized operators are well suited for a pyramidal scheme of multiresolution, and can also serve as oriented-edge operators. The Gabor scheme as suggested by Gabor [Gab46] is defined as the product of effective spatial extent and frequency bandwidth having minimum combined effective spread in the position-spectral plane, i.e. the product $\Delta\, x\, .\Delta\, u$ achieves the lower bound as compared to the joint entropy achieved by any other window function [PYZ88]. The

Gabor functions are essentially a special case of what is now known as the *Wavelets.* This is discussed later on in the chapter.

The Gabor functions have characteristics very similar to that of the Simple cells in human visual system. In the human visual system the cells are tuned to a particular frequency channel and orientation. A cell will fire only when it receives stimulus that corresponds to a particular frequency and is of a particular orientation. Similarly the Gabor filter functions exhibit detection features which respond to the *selected center frequency and orientation.*

In fact, when the Gabor functions are convolved with the image data, the processes of multiply, add, and threshold result in a decision. The decision is whether a *pixel displays a particular frequency and orientation, when considered as a part of the group of pixels* within the immediate neighborhood. If the pixel displays that frequency and orientation it is *fired.* This is analogous to the firing of a simple cell in the human visual system.

The window function $g(x, y)$ at x_0, y_0 may take on any form, and the integral is called a Window Fourier transform [GM89].

If one defines the window function $g(x)$ in one dimension to be a gaussian, it takes the following form :

$$g(x) = e^{-x^2}, \qquad -\infty < x < \infty$$

Therefore, for a window of the function $g(x)$ around x_0, one considers $g(x - x_0)$

$$g(x - x_0) = e^{-(x-x_0)^2}$$

Replacing $g(x)$ in the Window Fourier transform by $g(x - x_0)$ we get

$$Gf_{x=x_0}(u) = \int_{-\infty}^{\infty} e^{-\imath ux} g(x - x_0).f(x).dx$$

For any gaussian function

$$\tilde{g}(x) = g(-x) = g(x)$$

This means $g(x)$ is an even function.

The above is a Fourier transform of the Windowed function
$f(x).g(x - x_0)$
The fourier transform has a set of basis functions defined by
$e^{-\imath.u.x}$

From the above definition of the Window Fourier transform it can be said that the set of basis functions has been changed to
$e^{-\imath.u.x.} \times g(x - x_0) = g_{u_0,x_0}(x)$

Therefore, it is possible to define a filter function in terms of the inner product of the two functions $e^{\imath.u.x}$ and $g(x)$, where the inner product of the two functions $f(x)$, and $g_{u_0,x_0}(x)$ is defined as follows.

$$\int_{-\infty}^{\infty} e^{-\imath u_0 x}.g(x)f(x)dx =< f(x)g(x) >$$

When stated as a convolution this translates to:

$$g_{u_0,x_0}(x) * f(x)$$

If this window is a gaussian in the frequency domain it can be defined as:
$\hat{g}(u_0) = e^{-2.\pi^2.\sigma^2.u_0^2}$
So for standard deviation $\sigma_u = 1$ the window function is given by:
$\hat{g}(u_0) = e^{-2.\pi^2.u_0^2}$

Daugman uses the filter function as defined above (with a gaussian window in two dimensions). The kernel of the Window fourier transform in two dimensions is translated through a window to give a filter function defined on a window around (x_0, y_0) in the spatial domain, and (u_0, v_0) in the frequency domain [GD88].

5.4 Gabor Filter Functions

The Window Fourier transform that is arrived at after replacing the generalized window function $g(x)$ by the translated function $g(x - x_0)$, has a kernel that is the Gabor filter function. If the results in one dimension are extended to two dimensions, we get a filter function similar to the one defined by Daugman [GD88].

Hence, Gabor filtering of a signal in one dimension is a convolution of the signal function with a function like the kernel of the Window Fourier transform translated through a window $g(x)$ defined in the neighborhood of x_0. The kernel is an element by element product of a gaussian with a modulating function. This modulating function is a complex sinusoid. The modulating function takes on the form of Figure 6.3 in two

dimensions, which when multiplied with the gaussian function gives the Gabor Filter function of the type defined by Daugman.

This family of Gabor filter functions may be stated as [GD88]:

$$gf_{u_0,v_0}(x_0,y_0) = e^{-\pi[(x-x_0)^2.\alpha^2+(y-y_0)^2\beta^2]} \times e^{-2\pi i[u_0(x-x_0)+v_0(y-y_0)]}$$

The Fourier transform of the above equation is given by

$$\hat{g}f_{x_0,y_0}(u_0,v_0) = e^{-\pi[\frac{(u-u_0)^2}{\alpha^2}+\frac{(v-v_0)^2}{\beta^2}]} \times e^{-2\pi i[x_0(u-u_0)+y_0(v-v_0)]}$$

5.5 The Wavelet

Despite all its advantages, the Window Fourier transform, due to its fixed resolution and window size is not very convenient for the purpose of image processing.

The simple cell of the human visual system responds to a given size. This is to say that the number of variations in light intensity, and its spread in the visual space is a predetermined requirement for the firing of a cell tuned to that frequency.

The Gabor filter functions satisfy the frequency requirement. If the Gabor filter function coefficients are multiplied with a stimulus function, they behave like simple retinal cells. Analogous to the retinal cells, they fire in response to the center frequency of the filter. The number of changes in light intensity function per neighborhood of a cell (coefficient) is not the same for filters tuned to different frequencies.

For the human retina the number of changes in light intensity per retinal cell are the same. The frequency variation is achieved by changing the geometrical spread of these changes in visual space. The same number of changes in light intensity are dilated to give a low frequency tuning, and are contracted to give a high frequency tuning mechanism. Almost similar behaviour is displayed by wavelets as shown in Figure 5.5[1].

Gabor filter functions by their very nature have a drawback. They have a fixed resolution and size. In order to overcome these drawbacks another representation was introduced which is intermediate to the spatial and frequency representations—Wavelet transform [GM89].

Mallat has shown in his work that any function in $L^2(R)$ space can be represented as a decomposition on the Wavelet family of transforms. Conceptually, a Wavelet is a decomposition of a signal into a set of frequency channels having the same bandwidth on a logarithmic scale [GM89].

[1]This depiction of wavelets is as shown by Rioul and Vetterli [RV91]

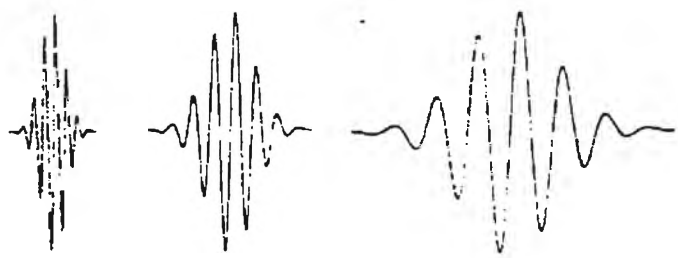

Figure 5.5: Wavelet : Same number of changes in light intensity function over differing geometric spreads.

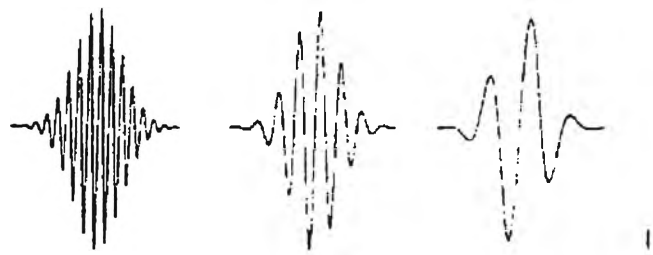

Figure 5.6: Gabor : Different number of changes in light intensity function fit into the same geometric spread.

Research workers have proved despite some reservations to the contrary, that some wavelets provide orthonormal basis of $L^2(R)$, and have wide ranging applications in a variety of fields [GM89].

Morlet first defined the wavelet transform [GM84] by decomposing the signal into a family of functions which are the translation and dilation of a unique function $\psi(x)$. The function $\psi(x)$ is called a wavelet and the corresponding wavelet family is given by $(\sqrt{s}\psi(s(x-u)))_{(s,u)\in R^2}$. The wavelet transform function $f(x) \in L^2(R)$ is defined by [GM89], [AGM89]

$$Wf(s,u) = \int_{-\infty}^{\infty} f(x)\sqrt{s}\psi(s(x-u)).dx$$

The idea behind the wavelet decomposition is not new. It is very much related to other types of spatial frequency decompositions, such as the Wigner-Ville transform.

A wavelet transform can be viewed as the filtering of $f(x)$ with a bandpass filter whose impulse response is $\tilde{\psi}_s(x)$. Unlike a window fourier transform which has a fixed resolution in spatial and frequency domain, the resolution of the wavelet transform varies with the scale parameter s. When the scale s is small the resolution is coarse in the spatial domain and fine in the frequency domain. If the scale s increases, the

resolution increases in the spatial domain and decreases in the frequency domain. This enables the Wavelet transform to isolate the irregularities of a signal at a local level.

Chapter 6

GABOR FILTER IMPLEMENTATION

"Vitality shows,
not only in the ability to persist,
but in the ability to start over".
– F. Scott Fitzerald –

6.1 Introduction

Visual systems in either humans or machines have to process massive amounts of image data that is generated by the input sources constantly. The human visual system and the vision system of some species of animals, higher up in the evolution ladder, manage to reduce the quantity of data that needs to be processed. This reduction is achieved by reducing the sampling rate depending upon the distance of the object from the human eye.

In order to achieve a reduced sampling rate, the human visual system employs what is known as position dependent sampling rate. This reduces the amount of data that needs to be processed for information [PYZ88].

Any scheme that hopes to simulate this property of the human eye must be capable of adjustable sampling rate, and localized filtering operation.

The filtering operation is carried out over a small localised area. This filter operator is to be repeated all over the image data space, filtering one small localised area at a

time.

The Gabor represenatation is well suited for this purpose (for a localised operator).

The two dimensional family of Gabor filter functions is defined as [GD88]:

$$Gf_{u_0,v_0,x_0,y_0}(x,y) = e^{-\pi[(x-x_0)^2.\alpha^2+(y-y_0)^2\beta^2]} \times e^{-2\pi\imath[u_0(x-x_0)+v_0(y-y_0)]}$$

The Fourier transform of the above equation is given by

$$\hat{G}f_{x_0,y_0}(u_0,v_0) = e^{-\pi[\frac{(u-u_0)^2}{\alpha^2}+\frac{(v-v_0)^2}{\beta^2}]} \times e^{-2\pi\imath[x_0(u-u_0)+y_0(v-v_0)]}$$

The above is a definition of a family of parameterized 2D Gabor filter functions as seen in Figure 6.4. They do the filtering of the image data in two dimensions, i.e., in the neighbourhood of point (x_0, y_0) in spatial domain, and for a frequency band defined over the neighbourhood centered at (u_0, v_0) in the spatial-frequency domain.

A general concept of the kind of operations that need to be carried out to implement these Gabor filter functions for the purpose of segmentation for multiresolution analysis of the image has been depicted in the structure chart of Figure 6.1

Figure 6.2 shows an individual element of the Gabor filter functions in the spatial domain[1]

One can also see the modulating complex sinusoidal functions in Figure 6.3 that are multiplied by a gaussian and then convolved with the image data to perform Gabor filtering at $x = x_0, y = y_0$, for frequencies centered at (u_0, v_0).

The paramenters α, and β are essentially scaling parameters that are used to vary the spread of the Gaussian function and tune its orientation.

If parameters α, and β are varied, such that $\alpha \neq \beta$, then a Gabor filter function is obtained. This filters the image in such a fashion that only those features in the image that are aligned with the direction/orientation of the filter function will be retained. The direction/orientation of the Gabor filter function can be seen in Figure 6.5.

6.2 Experimental Results

This project forms a part of the mobile robot program undertaken at the University Of Wollongong. A mobile robot Navigation system is already available in the Department. This Mobile robot has an onboard 68000 based controller. This controller generates the signals that drive the four wheels and control the direction and speed of the robot.

[1]All filter functions have been drawn using MATLAB, and the axes have been hand drawn in a few diagrams to give an idea of the parameters.

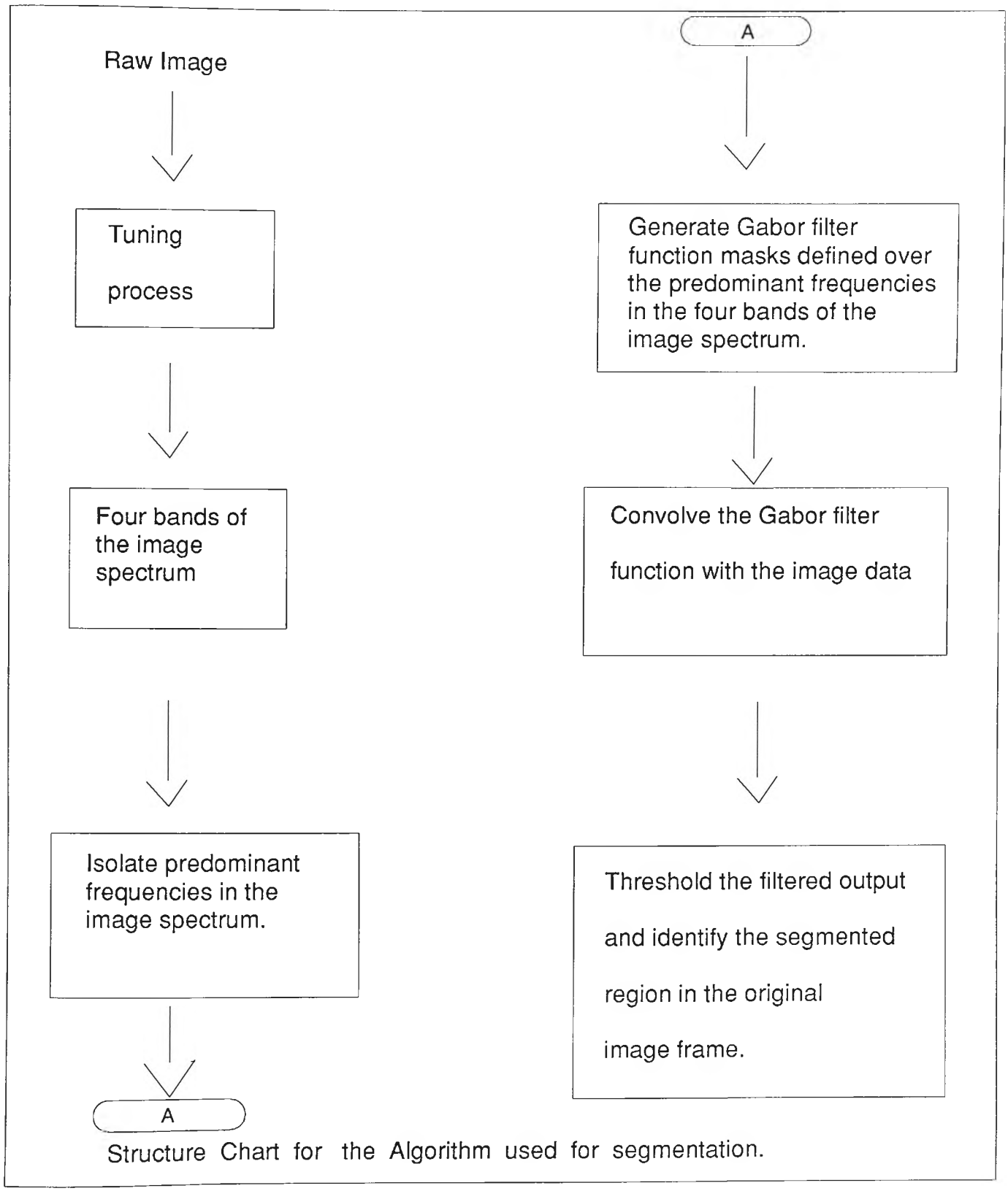

Figure 6.1: Structure chart of the operations carried out in the process of segmentation of the image

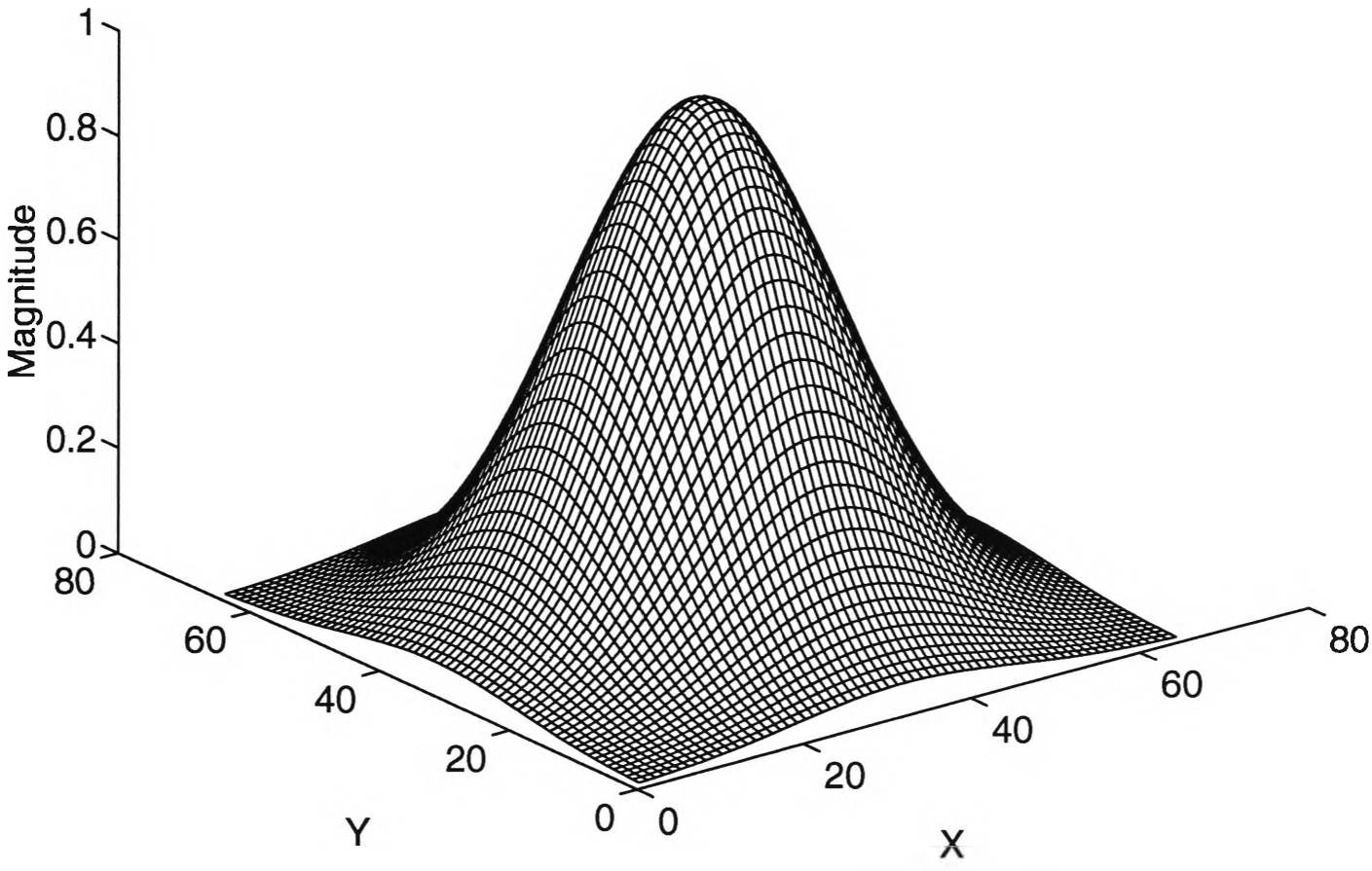

Figure 6.2: Gaussian function around point $x = x_0$ *and* $y = y_0$

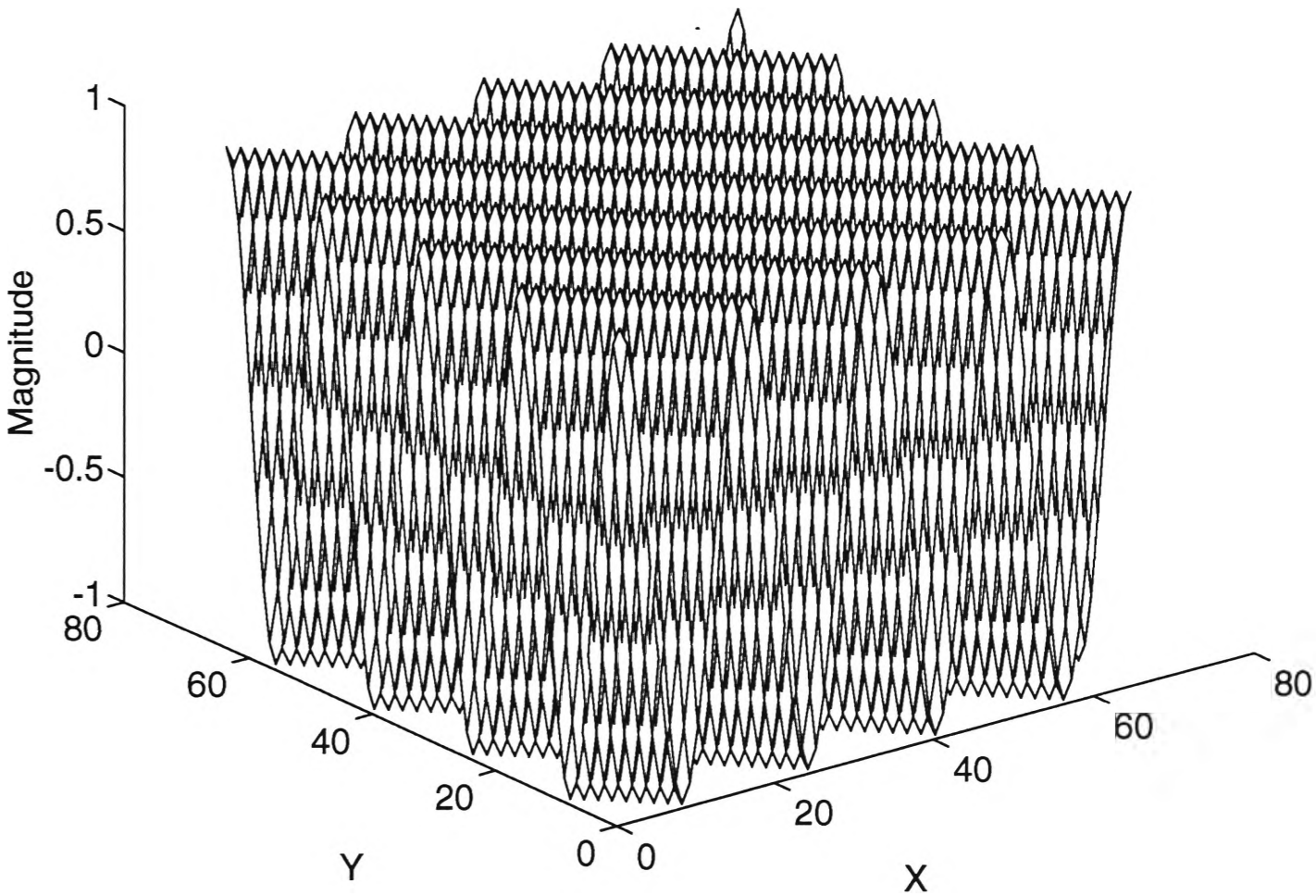

Figure 6.3: This Sinusoid function has a modulus of 1.

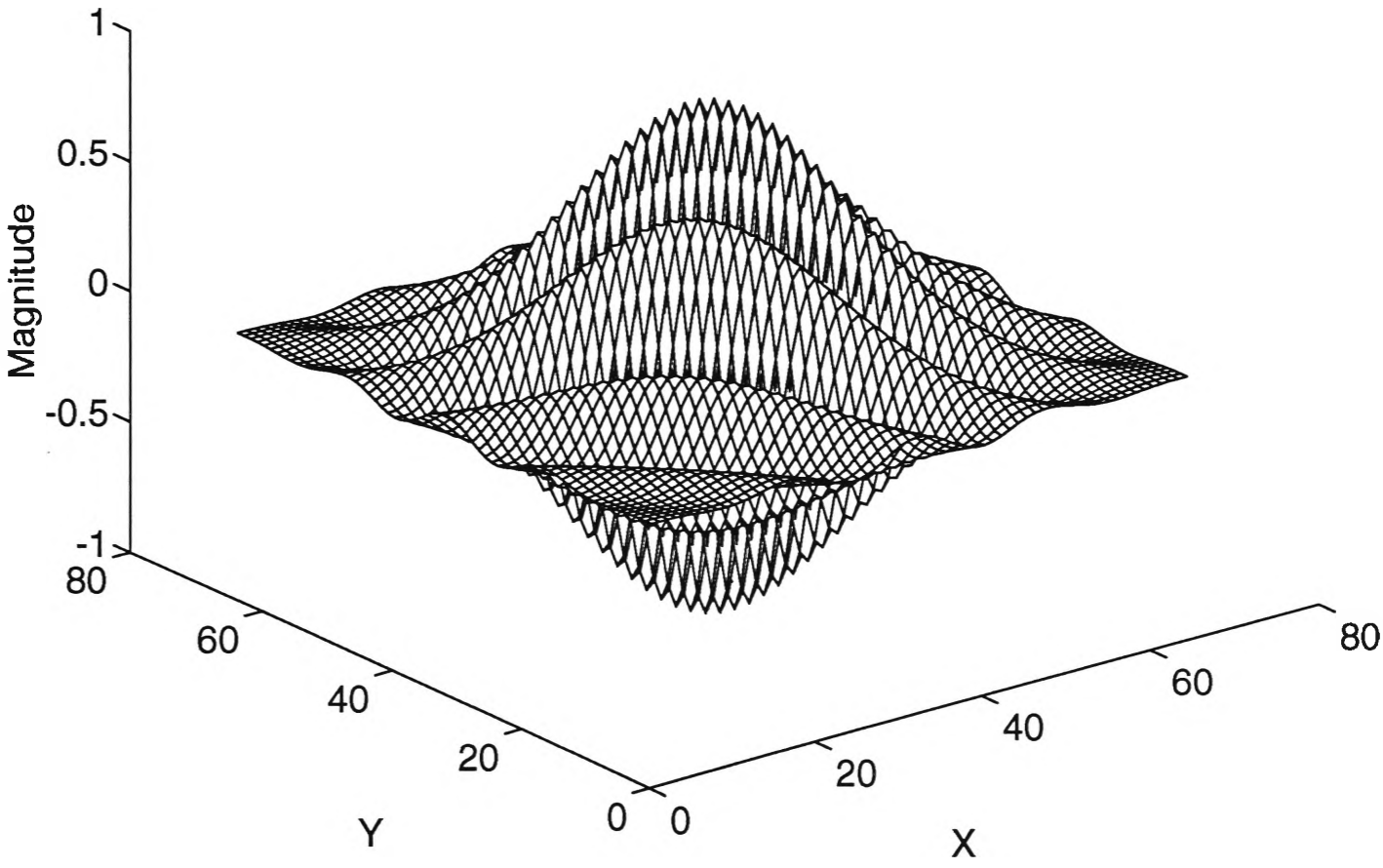

Figure 6.4: One of the Gabor family of filter functions in the spatial domain, that is a localized operator.

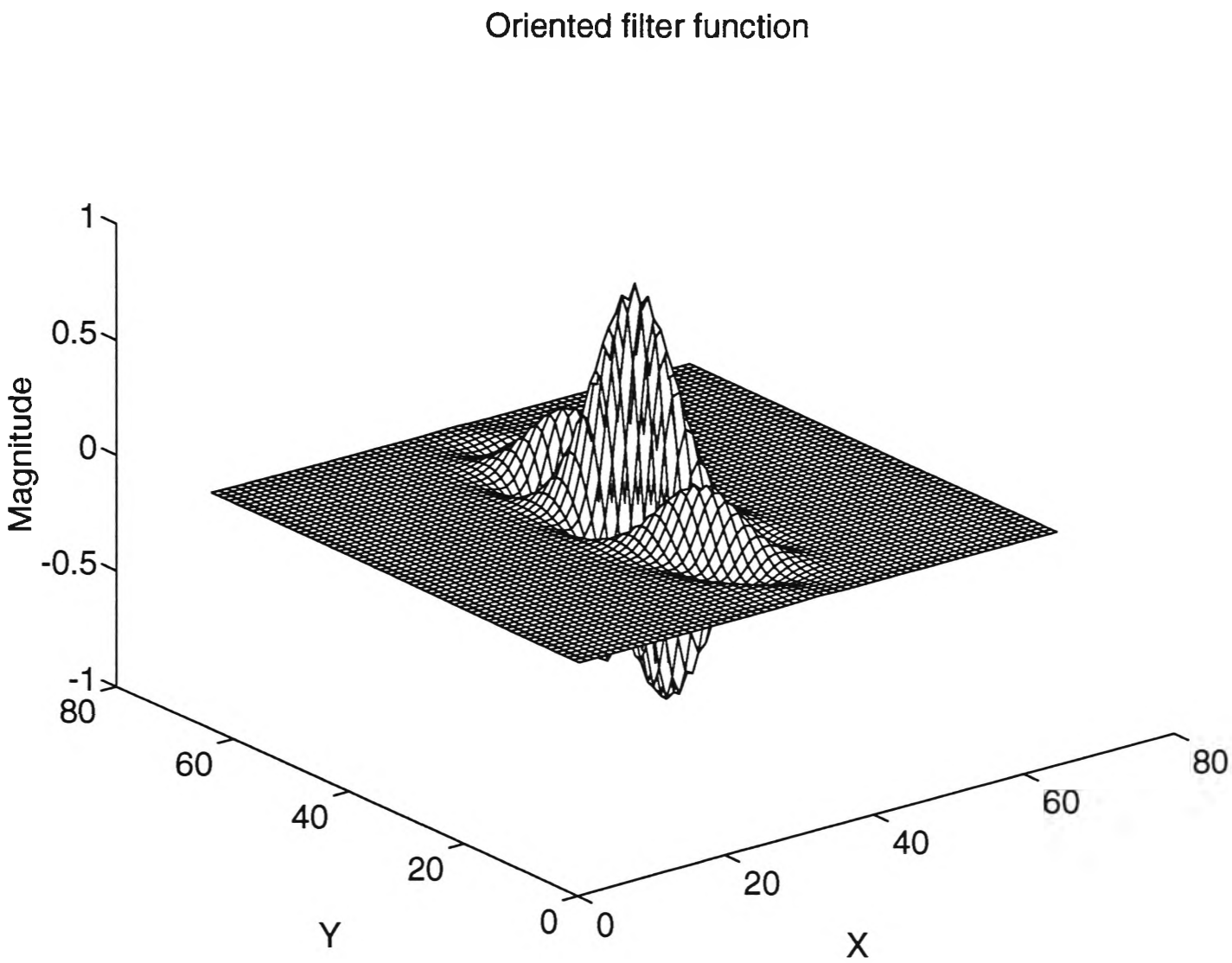

Figure 6.5: Plot of a Gabor filter function showing orientation.

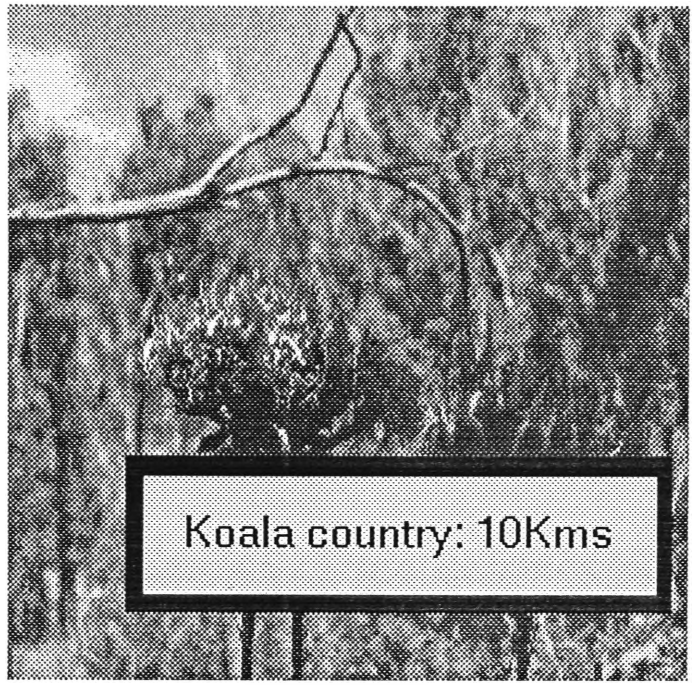

Figure 6.6: Signpost with text.

There is provision for the robot controller to receive commands from outside by way of hardwired connections, or by way of transceiver—*Telecommand module.*

The mobile robot vision system operates in a partially known environment. The natural working environment is sign posted by various text messages to facilitate the navigation of the robot. The primary requirement for this robot to be operational is to distinguish between different textures and segment them. Especially in the case of TEXT matter, the robot should be able to locate and segment the exact region that contains text matter in a given scene. If a sign post on a natural background, as in Figure 6.6 is considered, the robot should be able to locate the "Text" in the entire field of view, and scan it for optimal resolution. It should then pass it to a text interpreter to preprocess the textual information. Thus, the robot should be able to simulate the *"preattentive mechanism".*

Considering the problem of isolating the text written on a signpost with a natural background, it is necessary to localise the region of text which has a certain frequency. The spatial-frequency corresponding to the region of text (*area of interest*) is not known beforehand. For developing an algorithm that is sufficiently general purpose, the image has to be filtered through a localized operator.

6.2.1 Tuning Mechanism

The method adopted was to take a global Fourier transform of the image that is to be processed. This frequency spectrum in two dimensions is to be divided into 4 bands. Assuming that the image obtained by the camera fitted onto the Mobile Robot is going to capture images of size 256×256 pixels, the maximum spatial frequency that can be expected in that image is 128 changes/total sample, i.e. 128 cycles/sample.

This frequency range of 128 is divided into four bands of equal width having some overlap on both sides of the band. The next step is to find out the frequency at which the signal has the maximum amplitude/energy from all four bands (energy as defined in section B.1). These frequencies are then used as the central frequencies for the Gabor filtering of the images.

In any image of natural scenery, the Fourier spectrum is bound to have the maximum energy at DC or near-DC frequencies. The energy goes on decreasing until the highest frequency (half the Nyquist frequency). As a result, on considering the Fourier spectrum of an image, it is apparent that the amplitudes of the energy peaks fall off as one moves further away from the origin or the DC along the frequency axis.

To find the peaks in the spectrum the frequency domain representation of the image is run through the peak finding routine as detailed in Appendix C. Thus, the maximum values in all four bands in the Fourier spectrum of the image, and the frequencies at which these occur are found.

This algorithm gives the frequencies which display a high energy content in the Fourier domain. For two frequencies in the same band that are very close to each other and display high energy, Gabor filters can be defined, having those two frequencies as the center frequencies. This is possible because the Gabor transform uses the Gaussian as a Window function which allows a partial overlapping of two adjacent frequency spectra having the central frequencies that in turn have very small separation in the Fourier domain. The image frequency spectrum with all frequencies set to zero, except those that display high energy content, appears like in Figure 6.7.

For all the four center frequencies found by the tuning mechanism, Gabor filter functions are defined. These filter functions are convolved with the original image to obtain four versions of the original image. These four versions are the respective subbanded outputs of the original image for the selected center frequencies. The original image is filtered for localized spectral/textural variation that corresponds to the center frequency selected.

This operation of subbanding has no analogy in the human visual system since the human visual system is massively parallel. Those cells in the human retina that are tuned to a particular frequency, fire when a stimulus of the appropriate frequency is

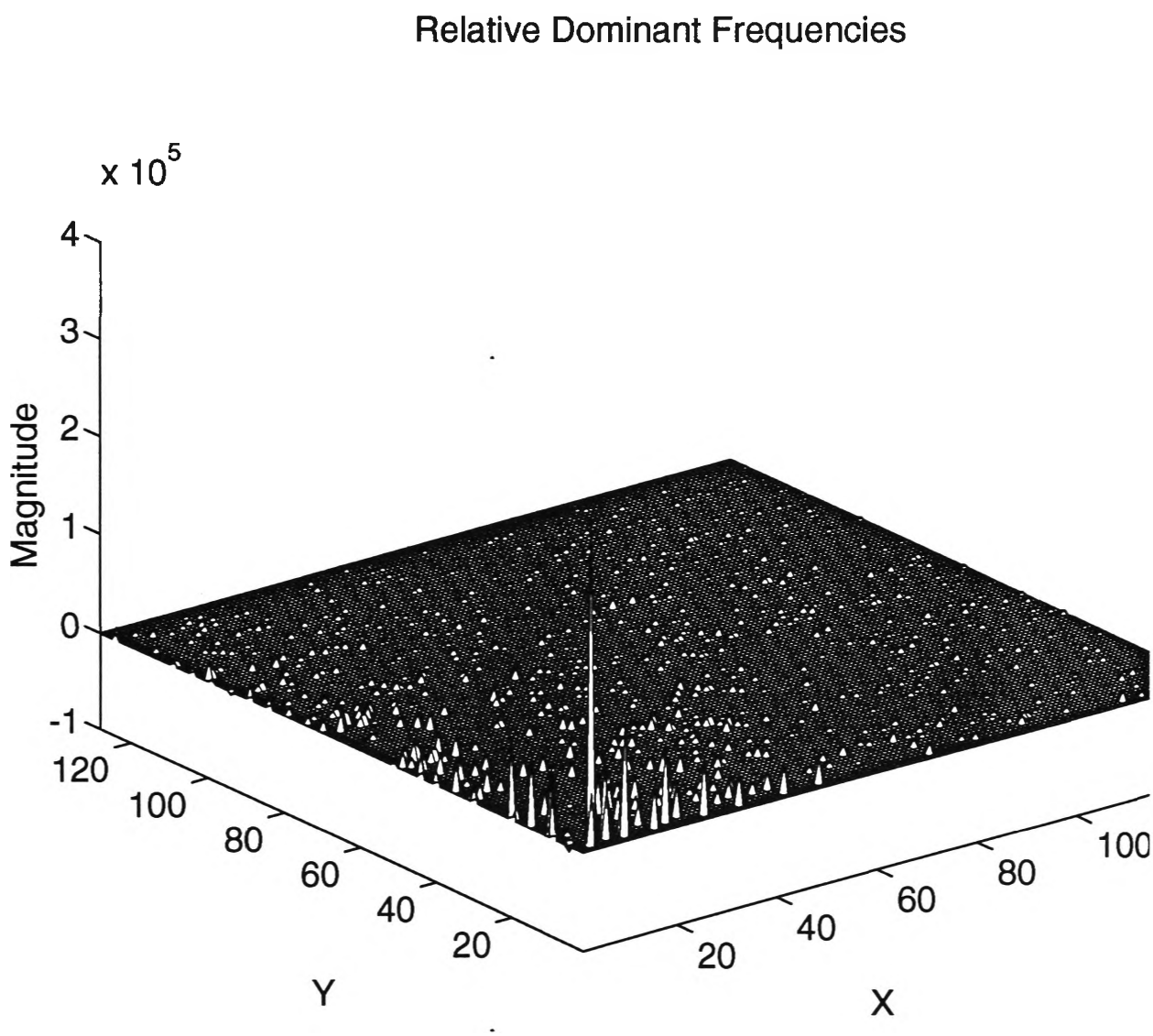

Figure 6.7: Frequency spectrum showing high energy frequencies.

Figure 6.8: UNICEF card used for filter experimentation.

given to them. The firing of the cells is a mutually exclusive event (two cells can fire simultaneously), in response to two different frequencies. The machine vision system for this project is based on a sequential machine. Hence, the need to subband the total spectrum and consider one band at a time.

Now these four versions or subimages of the original image may be thresholded to give a properly segmented output.

For the purpose of experimentation, two images were used viz. those shown in Figure 6.6 and Figure 6.8.

For the image in Figure 6.8, after the tuning mechanism has found the dominant frequencies, a filter mask is created and convolved with the image. A sample of the results obtained is shown in Figure 6.9 for a center frequency of $(u, v) = (50, 41)$, and standard deviation of $\alpha = \beta = 0.0625$.

6.3 Effects of variations in filter function parameters

If one were to consider the various parameters involved in the equation for the family of Gabor filter functions [GD88], the following variables would be considered.

- The center frequency of the filter function
- The standard deviation of the filter function along x and y directions

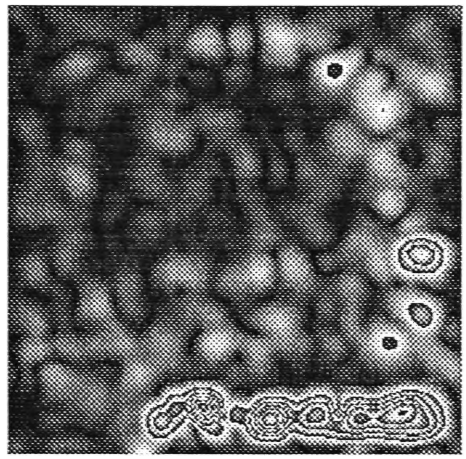

Figure 6.9: The image after convolution with the filter mask.

- The size of the filter mask as related to the spatial-frequency of the texture to be segmented. Spatial-frequency in this context means the relative size of the grains contained in the texture to be segmented.

6.3.1 Effects of variation in Standard Deviation

For studying the effects of variations in standard deviation of the filter function on the final segmented output, consider the image in Figure 6.6.

Under ideal conditions it is necessary to vary the filter mask-size to fit the **significant** coefficients of the filter function. Significant coefficients here mean those coefficients that have values V defined by

$$V > 0.1 \times (\text{maximum value of the filter coefficients})$$

It is not practically feasible to increase the mask-size infinitely, due to constraints on the computing power and the memory size available. The maximum size of the mask considered has been restricted to 33 $\times$ 33.

If very high frequency selectivity is necessary, and the center frequency of the filter function is near the D.C. region or the origin (low frequency), the filter function acquires a very large spread in the spatial domain and may not fit within the maximum size of 33 $\times$ 33.

The above mentioned considerations make it necessary to maintain the mask size at fixed value and vary the standard deviation of the Gabor filter function. Thus, by varying the resolution of the filter function in the spatial domain the effects are observed on the convolved image.

For a mask size of 33 $\times$ 33 the values of standard deviation are varied over a range

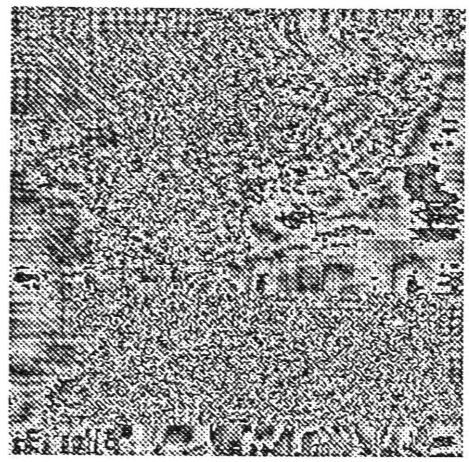

Figure 6.10: Filtered output for standard deviation=0.0325 and mask-size= 33 × 33.

of three sample values, viz. 0.0325, 0.0625 and 0.0925.

As can be seen from Figure 6.11 only the top part of the Gaussian shaped filter can be accomodated in a filter mask the size of 33 × 33. Since the filter mask contains only the top part of the total filter function, the filtering operation does not produce the expected output. This has been illustrated in Figure 6.10.

Before plotting the filter function for the other two values, i.e. 0.0625 and 0.0925, it will be beneficial to look at the relationship between the filter function in the *spatial domain* and the same filter represented in the *frequency domain.*

In order to obtain a better resolution in the frequency domain, it is necessary to increase the spread of the gaussian for the Gabor filter function expression in the spatial domain. The resolutions in the two domains, i.e. spatial and frequency are related to each other by the relationship stated in section 5.3.

The two inequalities are as follows.

$\Delta x.\Delta u \geq \frac{1}{4\pi}$

and

$\Delta y.\Delta v \geq \frac{1}{4\pi}$

In order to increase the resolution in any one of the domains, the resolution in the other has to decrease.

Now if the spread of the filter function is decreased and fits within the mask size, the filter function is better represented in the spatial domain. This is illustrated in Figure 6.12. Hence, the filtered output gives a much better segmentation as depicted in Figure 6.13, and compared to Figure 6.10.

For a further decrease in spatial spread of the filter function as in Figure 6.15, for standard deviation of 0.0925, a marginal improvement in the filter performance is obtained as shown in Figure 6.14.

The improvement in performance is marginal since even though the filter function

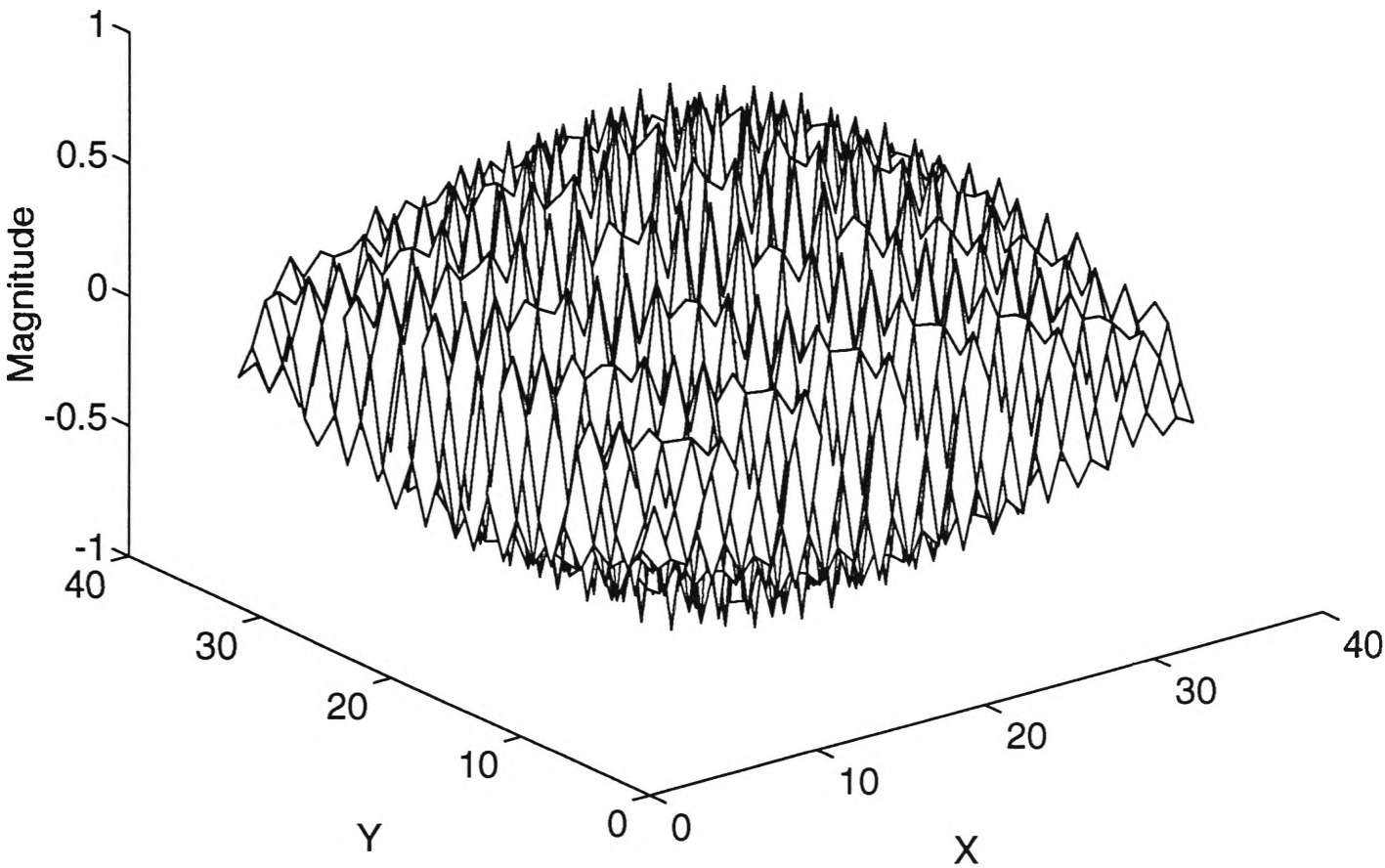

Figure 6.11: Gabor filter spatial form for standard deviation=0.0325.

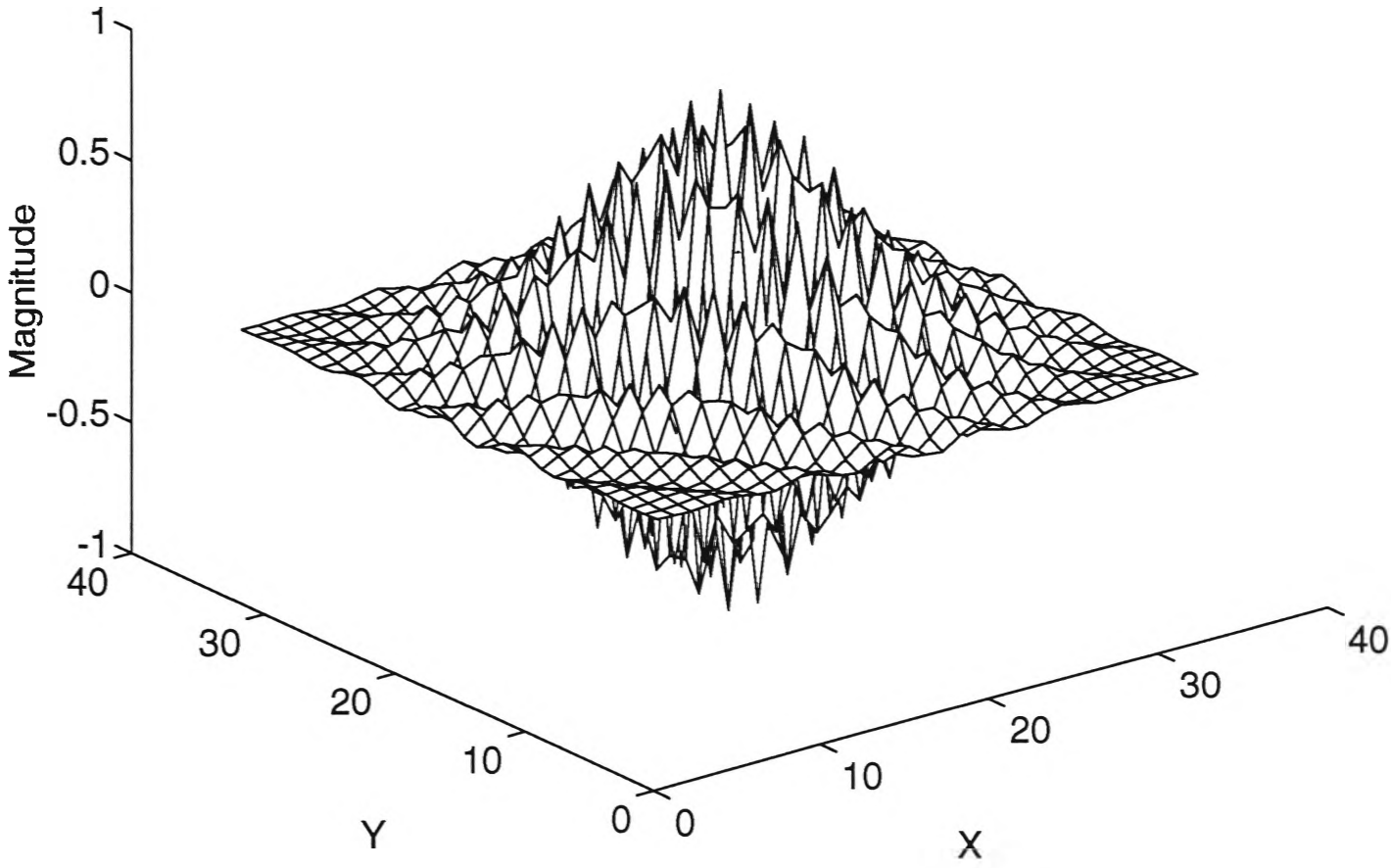

Figure 6.12: Gabor filter spatial form standard deviation= 0.0625,spread smaller than standard deviation= 0.0325.

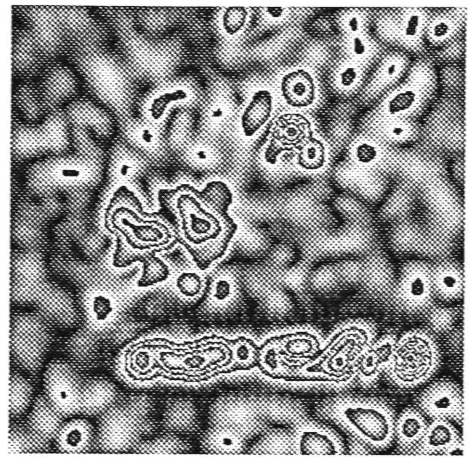

Figure 6.13: The filtered output for standard deviation of 0.0625

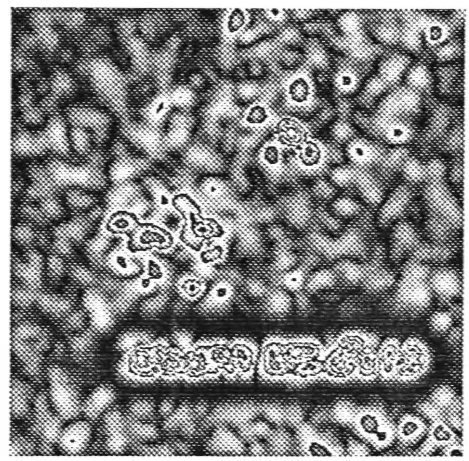

Figure 6.14: Marginal improvement in filtered output of text signpost on increasing spatial spread.

is adequately represented in the spatial domain, it has spread too wide in the frequency domain. Hence the filter function has a decreased resolution for the center frequency and the surrounding frequencies.

The above mentioned fact becomes evident if we compare the spatial representation of the filter for a standard deviation of 0.0925 as in Figure 6.15, to the frequency domain representation of the same filter as in Figure 6.16.

It is observed that both—the *spatial*, and the *spatial-frequency* representations of the filter, are gaussian functions. However, the above two representations have different standard deviations, and hence different resolutions in their respective domains. They are related to each other by the inequalities mentioned above.

It is evident that as the spread of the Gabor function decreases in the spatial domain, the spread of the function in frequency domain increases. That is, on increasing the standard deviation of the filter function in spatial domain, a narrow squeezed filter

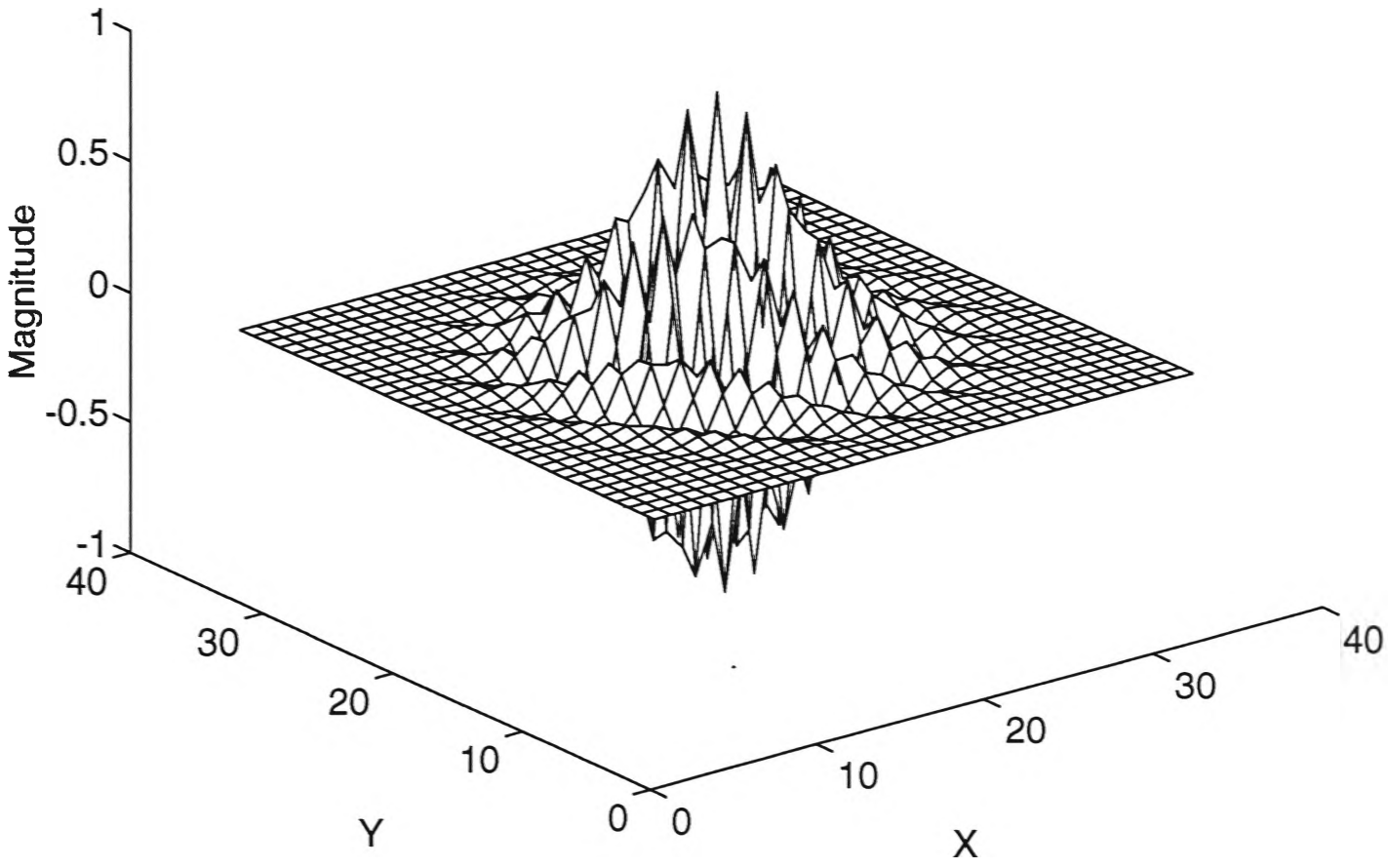

Figure 6.15: The Gabor filter function in the spatial domain representation, the spread in spatial is decreased.

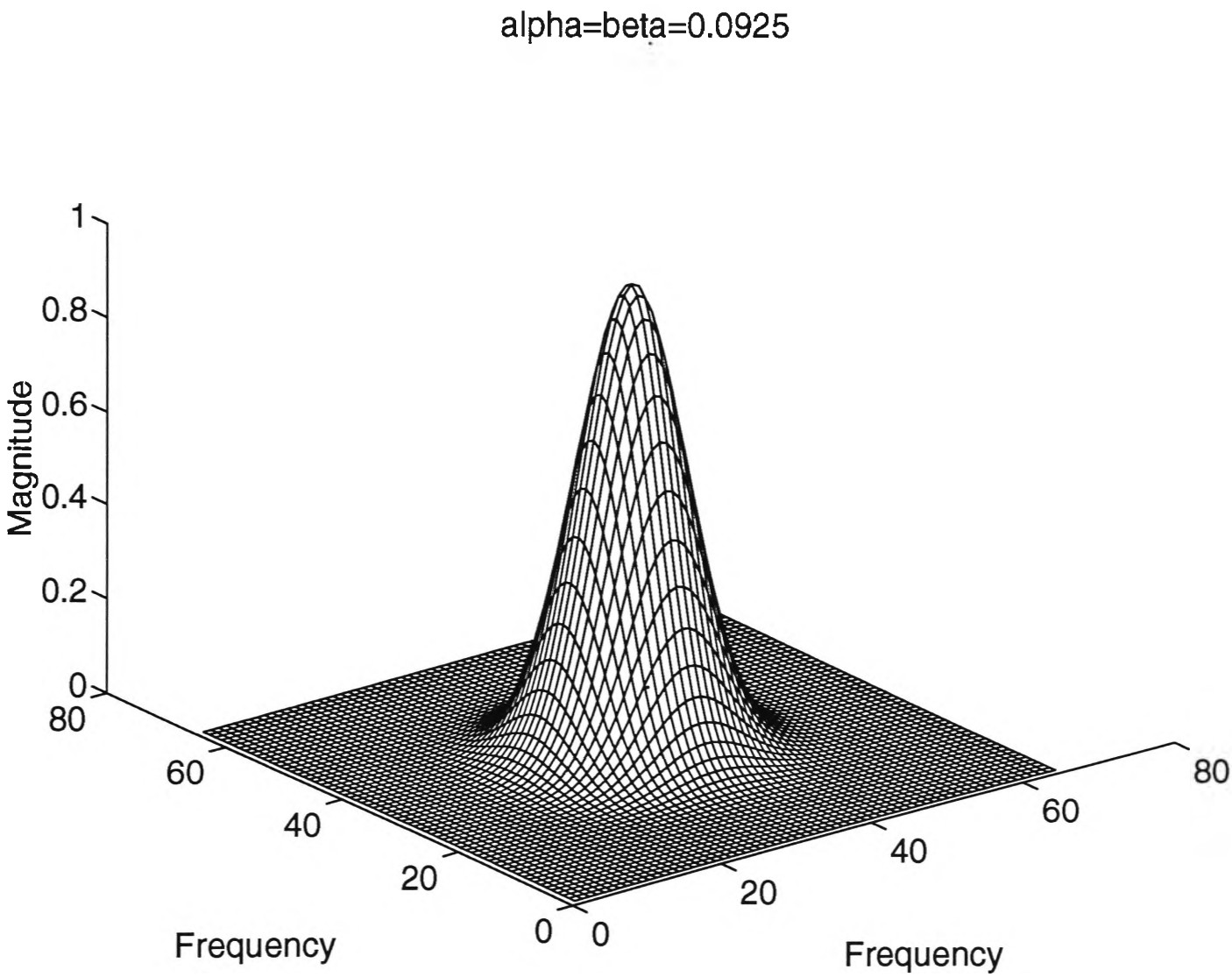

Figure 6.16: The Gabor filter function in the frequency domain representation, with increased spread.

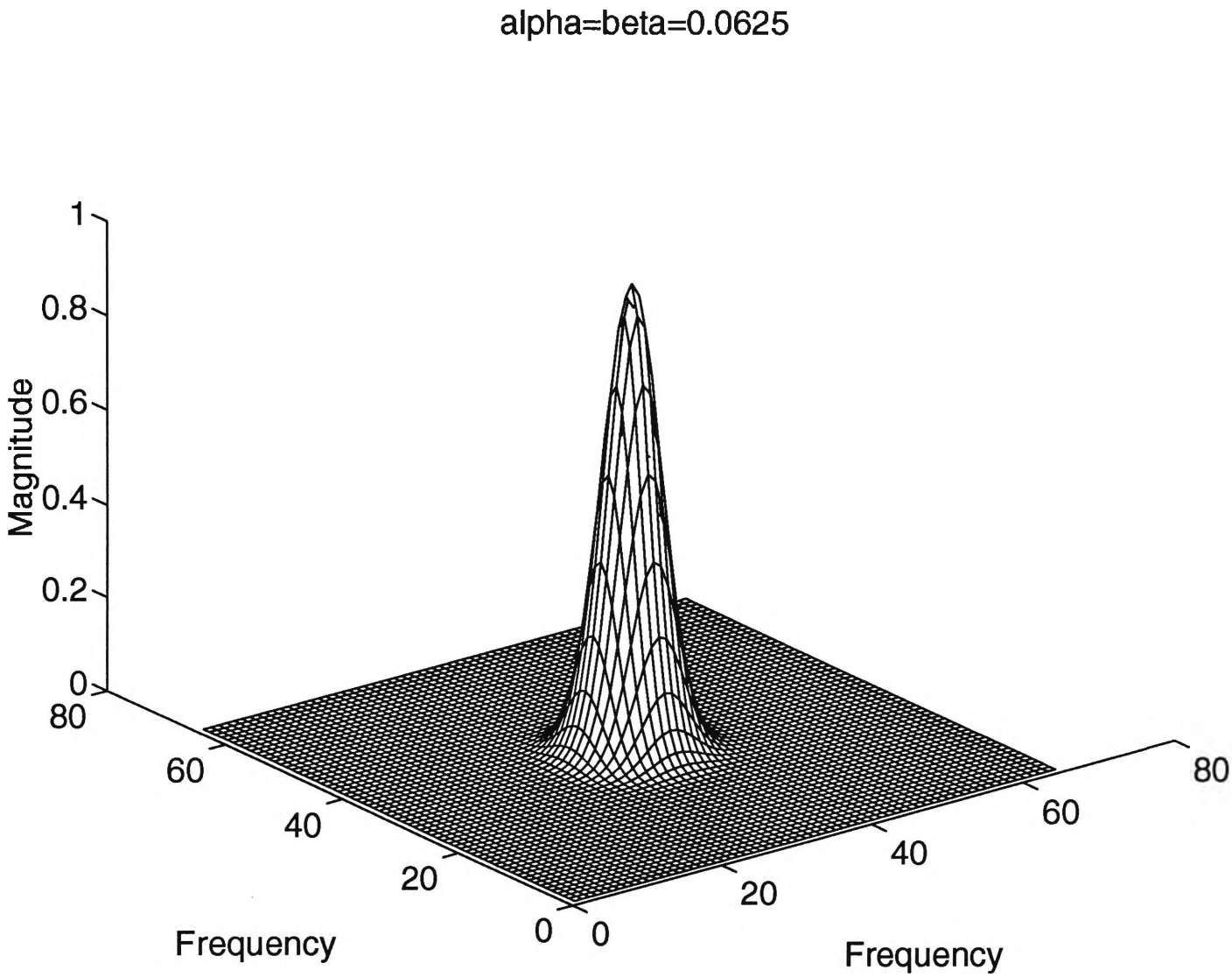

Figure 6.17: Filter function in frequency domain $\alpha = \beta = 0.0625$.

function in the frequency domain displaying a small bandwidth of frequency resolution or selectivity is obtained.

On comparing the frequency domain representation of the filter function for two values as shown in Figures 6.17 and 6.18, the standard deviation increases in the spatial domain from $\alpha = \beta = 0.0625$ to $\alpha = \beta = 0.0325$, and the spread in the frequency domain decreases.

6.3.2 Mask size variation

As mentioned earlier in Section 5.5 and Section 2.1.2, it is proposed to simulate the size recognition feature of the human visual system, which translates to spatial-frequency

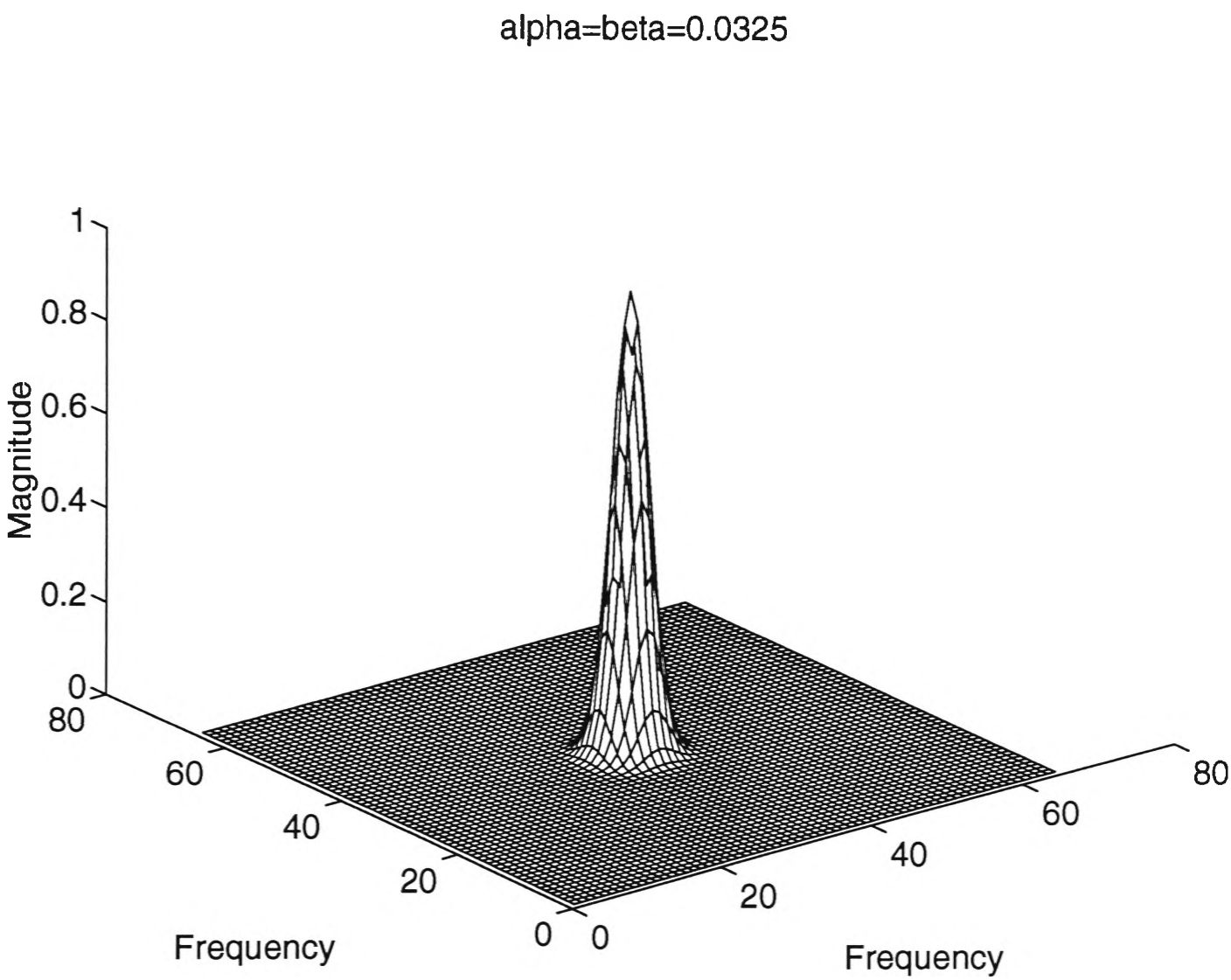

Figure 6.18: Filter function in frequency domain $\alpha = \beta = 0.0325$.

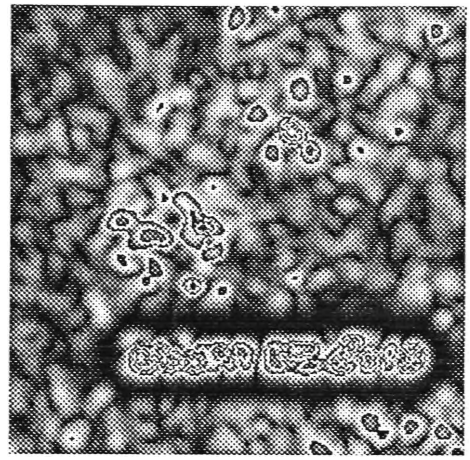

Figure 6.19: Standard deviation=0.0925 and mask size 25 × 25.

as a feature to be recognized in terms of machine vision.

To segment textures having fine grain (small size), consider a smaller spread in the spatial domain. A small spread in spatial domain translates to a window *"a mask"* of smaller size. Large sized feature segmentation translates to larger sized window.

The above results are based on simulating the human visual system, where every retinal simple cell responds only to a fixed number of changes in light intensity, but a different frequency (Section 5.5).

In order to simulate the same effect by way of varying the mask size and fitting in *approximately the same number of variations of the modulating function (*complex sinusoid*) within two standard deviations from the mean*, it is necessary to vary the standard deviation (*spatial spread*) so that the filter function fits within the mask.

The effective spread of the Gaussian weighting function can be taken as two standard deviations from the mean.

In the following few figures it will be evident that mask size and standard deviation for the Gabor filter function are directly proportional to each other. Theoretically it is advisable to vary the standard deviation and the mask size so that most of the significant coefficients of the filter function fit within the mask. Atleast those filter coefficients that fall within two standard deviations should fit within the mask to get a reasonable resolution in both the domains.

For a mask size of 25 × 25 and standard deviation= 0.0925 the filter function is too small for the masksize. Hence, for this masksize the standard deviation for this filter function in frequency domain will be large. The spread of the function in frequency domain will be large as in Figure 6.20 as compared to Figure 6.21 for the spatial domain. Thus reducing the filters frequency selectivity—leading to a lesser degree of segmentation as shown in Figure 6.19.

Now if the mask size is increased from 25 × 25 to 33 × 33, the standard deviation

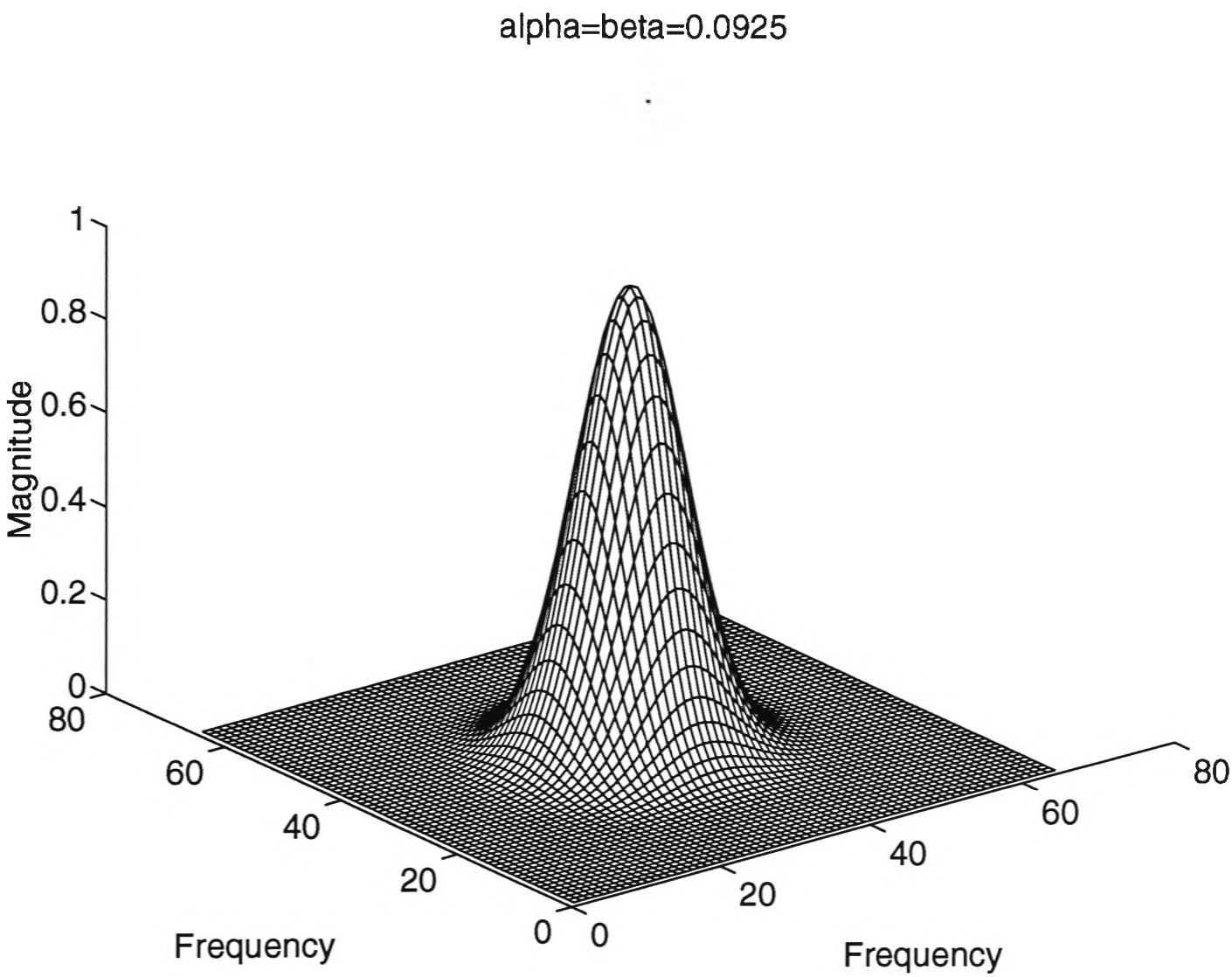

Figure 6.20: Frequency domain filter representation for standard deviation = 0.0925.

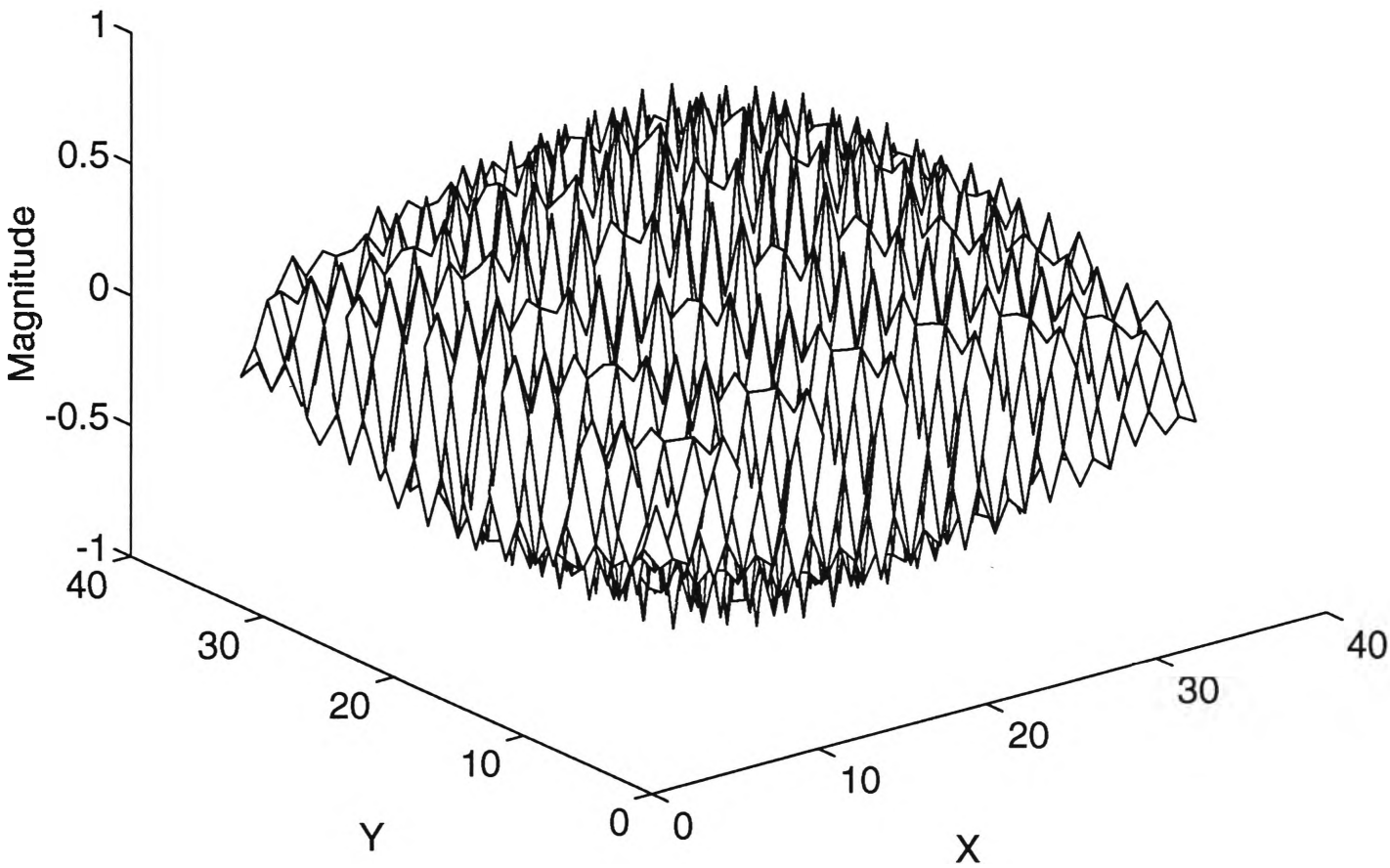

Figure 6.21: Small standard deviation and a large fixed mask size. .

has to be increased in proportion so that the filter function just fits in the mask size as in Figure 6.24. For a larger mask-size, as the standard deviation increases in the spatial domain, the spread of the function in the frequency domain reduces as in Figure 6.22. This gives a better resolution in terms of frequency selectivity as is evident from Figure 6.23.

As can be inferred from the previous discussion, the standard deviation of a filter function is tied with the mask-size, which in turn is dependent on the *grain-size (spatial frequency)* to be segmented.

The standard deviation in frequency domain indicates the frequency selectivity of the filter function, but this standard deviation is dependant upon the standard deviation in the spatial domain by the relationship given in Section 5.3.

Hence, a compromise between the values of α and β is necessary. Where α and β are the scaling parameters that determine the standard deviation along the two axes in the two domains.

6.3.3 Effects of variation in Threshold value

The filtered output may be subjected to binary thresholding. If the pixels gray-scale intensity value is greater than the threshold the pixel is set to white and if the grayscale intensity is less than the threshold the pixel is reset to black. This thresholding gets rid of the grayscale in the picture and a binary image is obtained.

For studying the effect of varying threshold levels on the final segmented output we consider the image in Figure 6.8

Consider the different threshold levels for the image in Figure 6.8. After filtering and thresholding for a pixel value of 100 the output obtained is as shown in Figure 6.25.

On comparing the Figure 6.25 with the original input image of Figure 6.8, it is found that at threshold value 100, a lot of extraneous output besides the desired *text information* is obtained.

On thresholding the filtered output of Figure 6.8 for higher values of 250, and 400 a better segmentation of the text matter is obtained. This is illustrated in Figures 6.26 and 6.27.

Since this project is applicable to a mobile robot in a partially known industrial environment, it is expected to see textures of varying sizes including text matter being used to indicate machine parts and/or instructions written on the wall. In order to enable the robot to scan the scenario and come to a conclusion that the field of view under scan at the moment contains two/three different texture patterns, it is necessary to filter the image over the total spatial frequency spectrum. The dominant frequencies are identified using the tuning mechanism suggested above. This will isolate for the

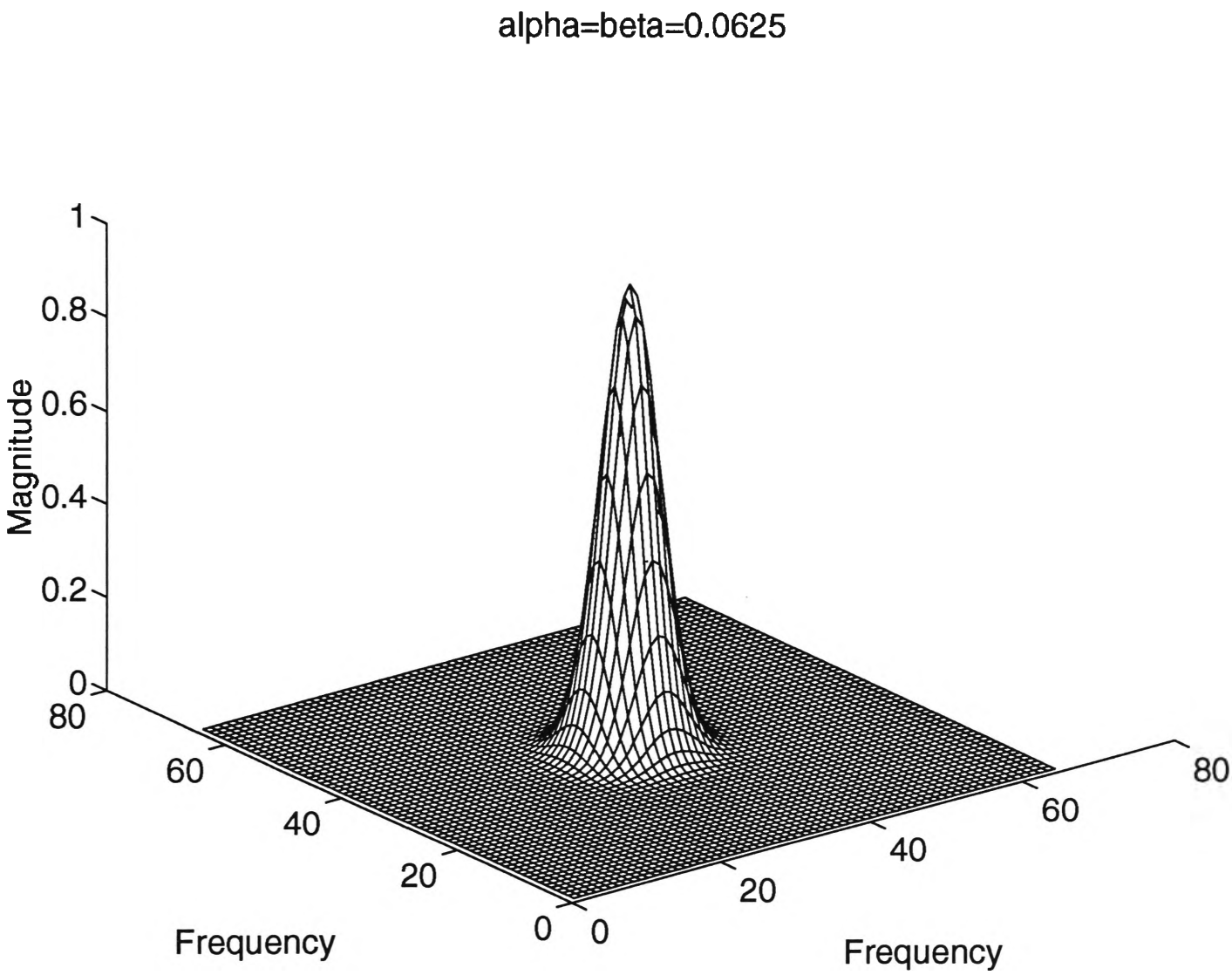

Figure 6.22: As the filter function spread increases in spatial domain, spread reduces in the frequency domain, $\alpha = \beta = 0.0625$.

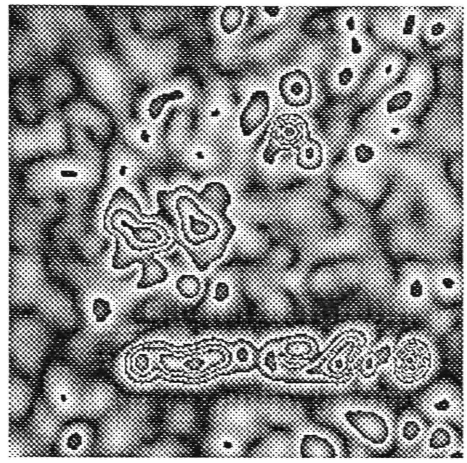

Figure 6.23: Standard deviation=0.0625 and mask size 33 $\times$ 33

robot the areas of interest. These areas show the local variation at the *dominant spatial frequencies*. Once the areas are segregated and the frequency at which the area was segmented is known, that particular region of interest may be scanned at the optimal resolution for that spatial frequency. The sampling rate can be optimised to extract the maximum information from the region of interest. In effect, achieving the simulation of the Human Preattentive Mechanism.

6.4 Segmentation of different sized text

An image containing two differently sized texture patterns with overlapping frequency content may be difficult to segment into the two exclusive textures.

A mechanism that recognises the size of the object to be segmented, besides the spatial frequency is required. Different resolutions can be obtained by varying the window size for different sizes of texture. The image on convolution with smaller window size will segment the high frequencies, and the larger window will segment the low frequencies.

In case of an image containing two different sizes of texture, the dominant frequencies are isolated using the tuning mechanism. The image is filtered for the different dominant frequencies.The following figures show this segmentation of two sizes.

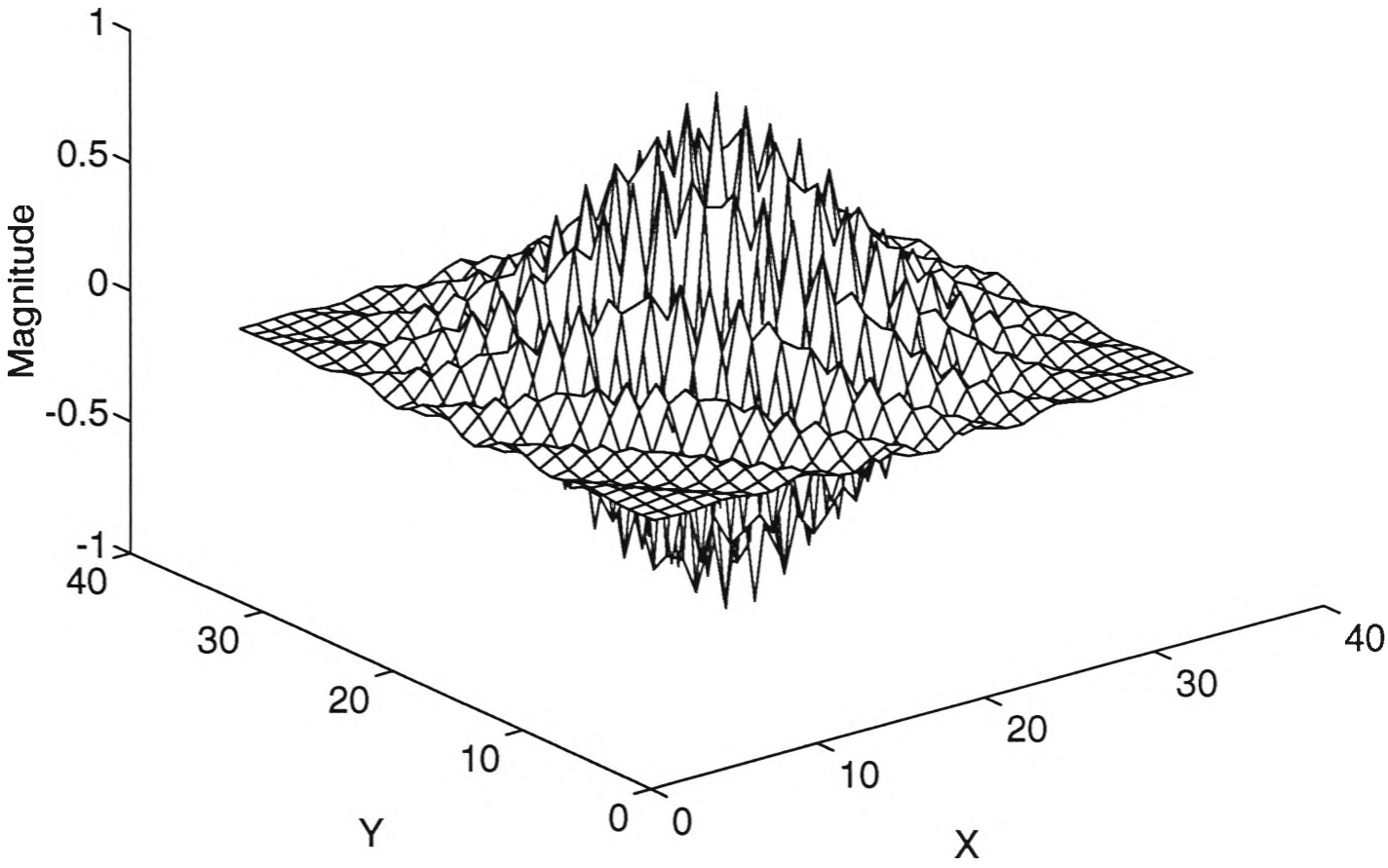

Figure 6.24: Filter spread such that it lies completely within the mask.

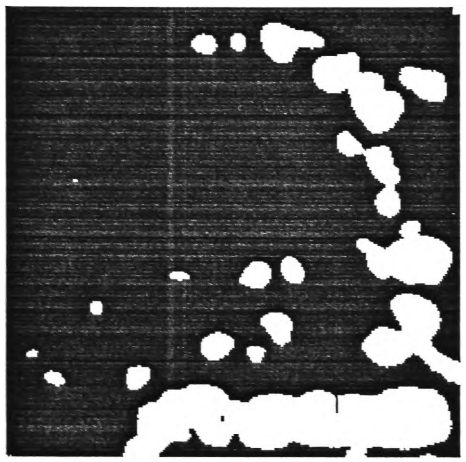

Figure 6.25: Thresholding for pixel value 100 for image of UNICEF card.

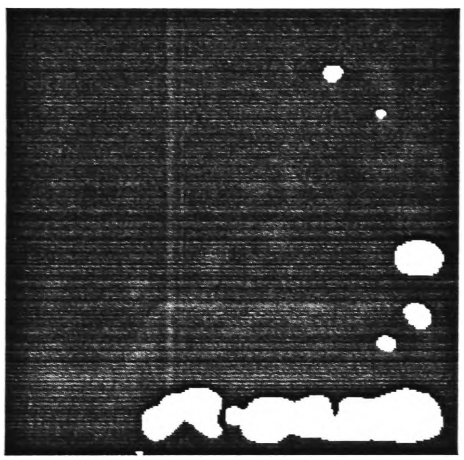

Figure 6.26: Thresholding for pixel value 250 for image of UNICEF card.

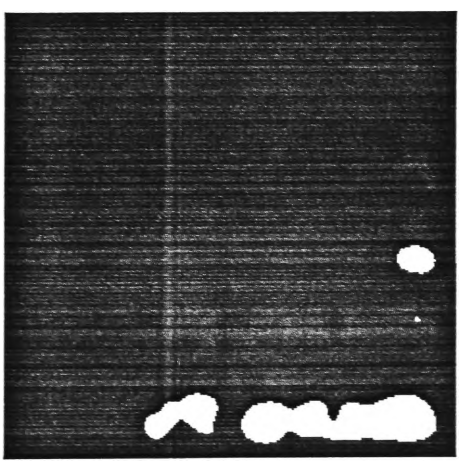

Figure 6.27: Thresholding for pixel value 400 for image of UNICEF card

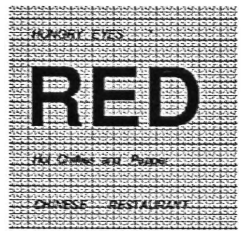

Figure 6.28: The image having two different text sizes

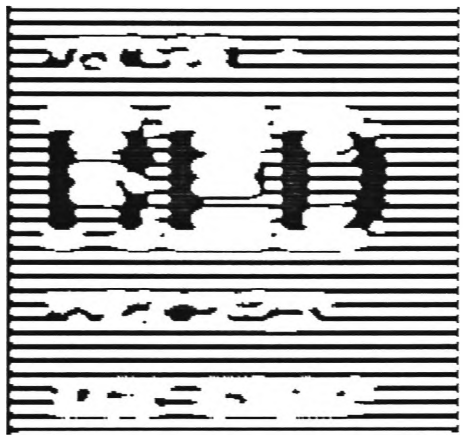

Figure 6.29: Image with two text sizes when filtered for a low frequency of (33,1) and thresholding gives us this output.

Figure 6.30: Image with two sizes when filtered for midrange frequency of (46,53), upon thresholding gives this result

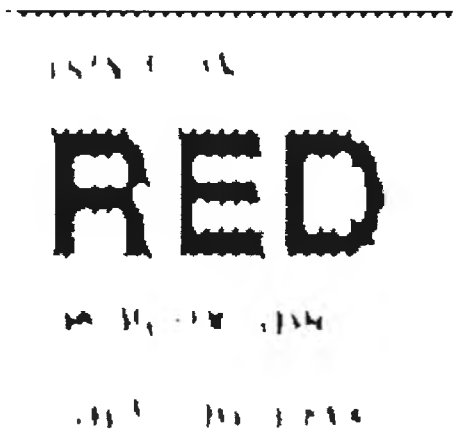

Figure 6.31: For very high frequencies and a small mask size the image with two sizes of text gives this result on thresholding.

Chapter 7

CONCLUSION

Everyone is trying to accomplish something big, not realizing that life is made up of little things.
– F. A. Clarke –

7.1 Review

The principal aim of this project has been the simulation of *Preattentive mechanism*, one of the inherent features of the human eye.

In computer vision, image decompositions essentially mean multiresolution. Most of the time the textures and structures that one would like to segment and then recognize have different sizes. Since one cannot know beforehand the optimal resolution that is necessary, a lot of methods for pattern matching at varying resolutions have been developed. The pyramidal structure [Ran91] for variable resolution is the most commonly used method.

The best known decomposition intermediate to the spatial and frequency representation is the Window Fourier transform. A special case of the Window Fourier transform has been used in this project i.e. the Gabor transform for filtering in the spatial domain. The principal concept of operating in the spatial domain instead of the fourier domain is centered around the expectation that this algorithm, with its one step procedure for texture segmentation, will be faster and easier to implement, especially on a parallel architecture.

7.2 Conclusion

The use of Gabor filter operator enables one to process any two dimensional signal to detect the presence of the desired spatial-frequencies, and if necessary the desired orientation. The texture in the area of interest has a corresponding spatial-frequency. Hence given an arbitrary frequency, the corresponding texture can be segmented.

To find out the spatial-frequency that corresponds to the desired texture one has to run the peak finding routine given in Appendix C.

Once the desired frequencies are determined the next step is to design the Gabor filter function as detailed in Chapter 6.

As discussed in Chapter 6 the variations in the parameters of the Gabor filter function allow the study of effects that the filter operator has, and analysis of its suitability for simulating the *Preattentive Mechanism* in the human visual system.

The Gabor filter serves the basic function of segmenting out the areas of interest. It satisfies the principal requirement as a tool for texture detection and segmentation. Since the resolutions in the two domains (*spatial and spatial-frequency*), are inversely proportional, it has its drawbacks.

As seen in Chapter 6, an image containing two different textures that display different spatial frequencies can be segmented using the Gabor filter functions having different standard deviations. But this has the following limitations. Consider two center frequencies $f(0)$ and $f'(0)$ for which the image is processed, are very close to each other (depicted in Figure 7.1). The Gaussian that is defined in a Gabor function around the center frequency f_0 will overlap the Gaussian defined around the center frequency f_0'. Thus the frequency spectrum spread of the two filter functions defined around f_0 and f_0' is not totally disjoint. The result being that when the image is filtered for one center frequency say $f(0)$, the presence of the other frequency $f'(0)$ will be apparent in the filtered output. However the energy displayed by f_0' in the filtered output will be lower than that of f_0. This indicates that the Gabor filter operator is inefficient in selecting frequency when "spatial resolution" of a very high degree is required for texture detection and segmentation.

The minimum separation between the two frequencies in the frequency domain is governed by the uncertainty relationship given by the following two equations

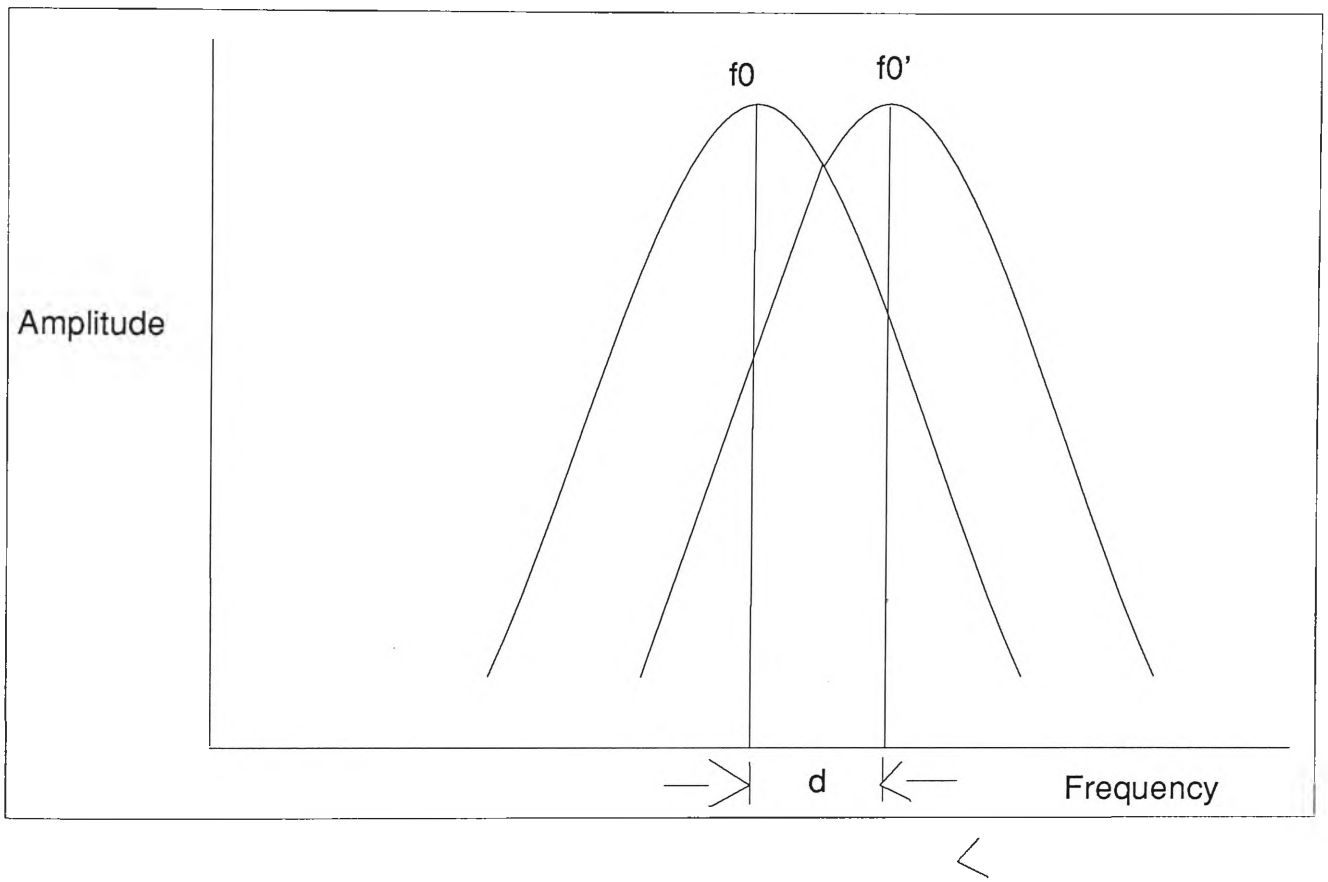

Figure 7.1: The center frequency in a Gaussian function gets the maximum weight, as a result frequencies close by can be differentiated.

$$\Delta x.\Delta u \geq \frac{1}{4\pi}$$

and

$$\Delta y.\Delta v \geq \frac{1}{4\pi}$$

On the other hand this is an image representation that affords some degree of separation from the neighbouring frequencies. Considering a Window Fourier transform having a square or Boxcar shaped window, it would not be possible to segregate those frequencies that are close to each other. The disadvantage of using a Boxcar type of window is that it does not achieve the optimal resolution in the conjoint spatial and spatial-frequency domain, i.e., it does not achieve the lower bound of the inequalities stated above. Even if one considers extremely small Boxcar shaped windows around two frequencies $f(0)$ and $f'(0)$ one finds that on taking the Fourier transform the resulting *"sinc"* shaped window does not allow achieving the lower bounds in the inequalities given above. The same applies to any other shaped window. The Gaussian window is the only one that satisfies these two inequalities.

In a situation where very high spatial resolution is required and simultaneously frequency selectivity of a high order is also a must, the Gabor filter operator does not serve the purpose. What is needed is another spatial/spatial-frequency representation wherein the resolutions in the two domains are separable if not mutually exclusive. These resolution parameters should be able to control the resolution in both domains irrespective of each other. This enables one to manipulate the resolution of the feature detection tool to one's satisfaction.
During the course of this project it was observed:

- For very high frequencies the window size, (determined by the smallest frequency one wants to segment), if reduced beyond a certain limit (depending upon the pattern of the texture), failed to give the expected results [HCW92].

  The above observation gets rigorous mathematical support from Janssen et al [Fol86], [Jan81] when they observe that, *"for analyzing high frequencies, where rapid variations take place over small regions, it is better to use wave packets that are more tightly localized in position space. And Wavepackets of the* same shape and fixed size *do not work well as a basis for $L^2(R)$ space"*.

7.3 Further work

The drawbacks of the Gabor transform can be overcome by the Wavelet transform, which does not have a fixed resolution, as in the Gabor transform. This alternative was not tried out during the course of this project.

Generally the structures that one wants to recognize have varied *grain-sizes (spatial-frequency)*. Defining an optimal resolution is not possible beforehand because of this difference in sizes. The Wavelet transform as explained earlier in Section 5.5, has a variable resolution and size detection feature [WCP92].

A wavelet transform decomposes the signal into a set of frequency bands having a constant size on a logarithmic scale [GM89] [WADCD92] [WCP92].

The FFT has basis functions which are the familiar sines and cosines. The wavelet transform has basis functions which are called *wavelets.*

The difference is that unlike the sines and cosines of the Fourier transform the wavelets are localized in space; and simultaneously like the sines and cosines, individual wavelet functions are localized in frequency or what is termed as characterstic scale.

Unlike the Sines and Cosines of the Fourier domain that give one a unique Fourier Transform, the wavelets are numerous and generate infinite transformations for the original signal depending upon the choice of wavelets [Pre91].

There is a trade-off in the selection of the different sets of wavelets, between how compactly localized and smooth they are in space.

As a future course of action, a detailed study of *wavelets* and implementation of a discrete version of the wavelet transform, will enhance the understanding of the functioning of some of the features of the human visual system. A comparison of its performance with other algorithms used in computer vision may prove quite fruitful.

Appendix A

THE HUMAN EYE

"How much more admirable the Bhagvad-Geeta, than all the ruins of the 'East'!"
– Thoreau –

The general organization of processing in the eye-brain system in humans is indicated in figure A.1 [Bro86]. The scene is sensed by the two eyes. A lens in each eye focuses an image of the visual field onto the retina. A network of cells of two types in the retina, called respectively "rods" and "cones", sense light intensity. They generate electrical signals comprising bursts of impulses whose frequency is less than $1000Hz$. The number of impulses in a burst is proportional to the time rate of change of light intensity. Rapid small motions of the eyeball (called *saccadic* motions) ensure that even a stationary scene produces a time-varying stimulus, and hence generates bursts.

These signals are processed further within the retina in so-called X-cells in which groups of excitatory sensors are surrounded by rings of inhibitory sensors; the latter tend to curb the activity of the former. The result is that the bursts of impulses generated by the X-cells depend also on the spatial rate of change of intensity of the retinal image. The effect is to filter the image with a band-pass filter whose response is circularly symmetrical. This response is a *difference of Gaussians* (DOG) form, which is very close to a doubly-differentiated Gaussian response. The consequence of this spatial filtering is to enhance edges (i.e., places where the rate of change of intensity with distance is high), without regard to the orientation of the edge.

The partially-processed signals leave the retina via a bundle of nerve fibers at the

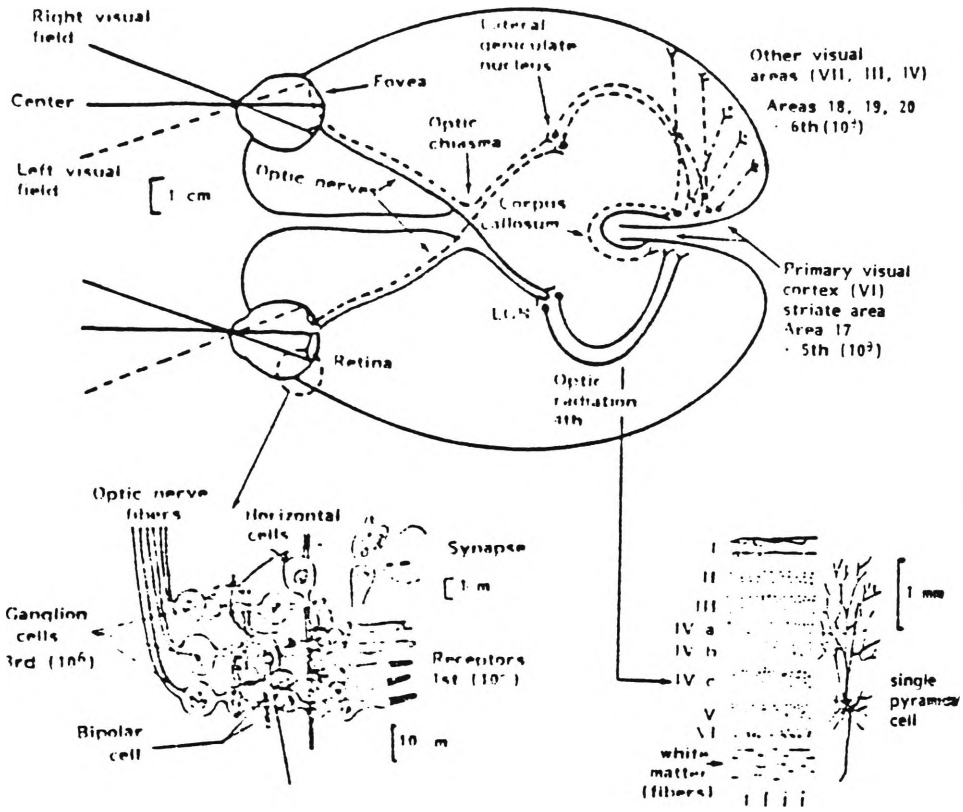

Figure A.1: Visual Processing in the Human Brain, viewed from above.

"blind spot". This bundle meets that from the other eye at the optic chiasm; half the signals from each eye are processed in each side of the brain to achieve stereoscopic vision. The visual signals are analyzed in the rear of the striate cortex in the cerebrum of the brain. Groups of cells present here are sensitive to edges of different orientation, both fixed as well as moving in particular directions. The outputs of these (and other cells, sensitive, for example, to general illumination level and colour) are combined and processed further, for example , to analyze scenes and identify objects. This high-level processing is, as yet, far from understood.

It must be stressed that some information regarding the positioning of objects within a scene, and hence of depth, is obtained from single images using clues such as occlusion (Some parts of the scene are partially hidden by others) and relative size. The processing required to extract and interpret these clues is extremely complex. In contrast, that required for depth perception using stereo vision operates even in the human visual system at the lowest level; it will work even with random textures.

For visual automation purposes, the most immediately interesting characteristics are those of the eye itself. Whereas man-made electronic cameras have only a single output channel and must transmit their data serially, the eye has hundreds of thousands of channels emitting in parallel though at a relatively low rate. The eye-brain system

has a response time of about 1/30 sec.

When examining a scene, the viewer moves his eyes so as to scan with his foveal region small regions of the scene containing the information needed to further his analysis. Thus, the regions scanned depend on exactly what the observer is looking for, and a particular region may be scanned several times as the analysis proceeds. This arrangement nicely accommodates a weakness of the eye (compared to a man-made camera), in that aberrations due to the lens in the eye are very severe and a sharp image is obtainable only within a cone about 1° wide around the optic axis. In contrast, a man-made lens can have superb image quality over a field of 20° or more. The scanned search process is efficient and economical, information is not taken into the eye-brain system until and unless it is evidently needed.

The eye focuses automatically to cover a field extending from 25 cm to infinity. Spatial resolution is essentially an angular quantity, thus it is maximum in distance terms at the near limit of the field; here it is about 70 microns for the foveal region used for critical work. Resolution may be specified in the space domain, as an ability to perceive that two dots which are close together are, in fact, distinct. The same applies to the spatial frequency domain. This latter involves specifying its attenuation of pure sinusoidal patterns at various spatial frequencies. The variation of attenuation of the amplitudes of the patterns with frequency is termed the modulation transfer function (MTF). Generally, attenuation increased with spatial frequency. The eye is most sensitive to spatial frequencies of about 6 cycles /1°.

The human eye is a remarkable instrument with respect to both sensitivity and resolution. In clear air, a candle flame is just visible at a distance of ten miles, thus, 10–14 parts of the light produced by a single candle is sufficient to stimulate vision. The mechanical energy of a pea, falling from a height of one inch, would, if translated into luminous energy, be sufficient to give a faint impression of light to every person that ever lived. Some of the parameters of the human visual system are:

- 120 million rod cells in each eye.
- 6 million cone cells in each eye.
- 2000 cone cells in each fovea in the region of maximum uniform density.
- 1 million nerve fibers in the optic nerve exiting each eye.
- Diameter for cone cells in fovea: 1 to 3 micrometers.
- 250 million receptor cells in the two eyes vs. 250 000 independent elements in a TV picture.

- Distance from effective center of lens to fovea: 17 mm
- Interpupillary distance : 50 to 70 mm
- Visual angle subtended by fovea: 20 minutes of arc for region uniform maximum cone density, 1 to 2 degrees for rod-free area, 5 degrees for a 50 percent drop in visual resolution (with the arm extended, the raised thumb subtends and angle of 2 to 2.5 degrees; one minute of arc corresponds to a retinal image of five micrometers).
- Angle with respect to visual axis of eye at which rod density is maximum : 15 to 20 degrees.
- Rod cells are on the order of 500 times more sensitive to light that cone cells.
- Visible portion of the electromagnetic spectrum: 0.4 to 0.7 micrometers.
- Wavelength of maximum rod sensitivity: 0.51 micrometer (green).
- Wavelength of maximum cone sensitivity: 0.56 micrometer (orange).
- Intensity range: 10 16 (160 decibels).
- Minimum visual angle at which points can be separately resolved: 0.5 to 2 seconds of arc for alignment of lines (0.04 to 0.16 micrometer, less than 10 degrees of the diameter of the smallest foveal cell); 10 to 60 seconds for dots (range of values is due to disagreement across reference sources). If a 0.55 micrometer light is used, central circle of 3.7 micrometers of the retina. This would mean that the illumination from two points 25 to 30 seconds apart would overlap.
- Object distance from eye for stereoscopic depth perception : 10 inches to 1500 feet (1500 feet corresponds to a retinal disparity of approximately 30 seconds of arc)
- Involuntary eye movements: 10 to 15 seconds of arc for tremor; slow drifts of up to 5 minutes of arc.

How then is the information from the eyes coded into neural terms, into the impulses of the brain, and then interpreted? When light strikes the retina, the decomposition (bleaching) of pigments in the rods and cones result in electrical activity, which is integrated in the bipolar and ganglion cells comprising the sixth and eighth levels of the ten-layer system of the retina. As discussed, the ganglion cells of the eye feed the brain with visual information coded into chains of electrical pulses. The rate of "firing"

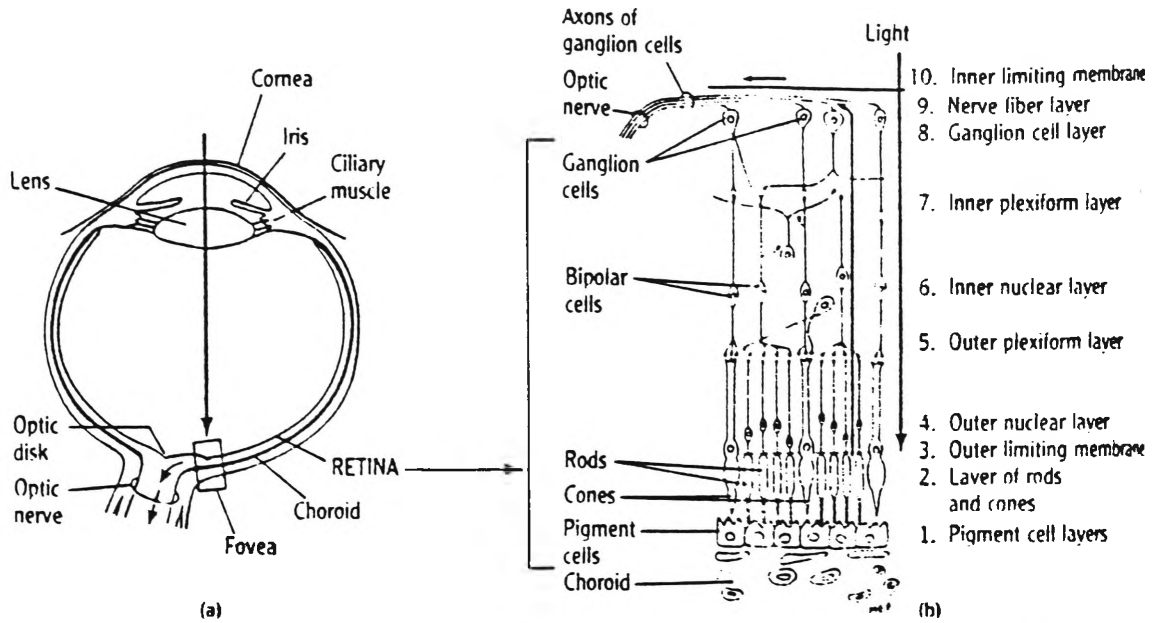

Figure A.2: Anatomy and the nervous organization of the human eye.

of the cells is proportional to the logarithm of the intensity of the original stimulation (Fencher's law).

Other attributes of the illumination, such as colour, are determined by which cells are firing. For example, as indicated in the figure A.2, cone cells are differentially sensitive to the red, green, and blue components of the illumination because of differences in the chemical composition of their photosensitive pigments. These cells are intermixed in the fovea, and their relative excitation provides the brain with information about the colour of the objects being viewed.

As depicted, the human retina in figure A.2 is effectively divided vertically down the middle; the nerve fibers from the left half of each retina send information about the right half of the visual field to the striate cortex in the left occipital lobe of the brain. Similarly, the right half of each retina sends information about the left half of the visual field to the right striate cortex. The role played by the lateral geniculate body is not currently understood, it appears to simply relay the information it receives. (However, there is some evidence that it is functionally involved in the processing of colour information).

The nerve fibers from the eye, reaching the striate cortex, preserve the topology and much of the geometry of the imaged scene information; a portion of the striate cortex.

called the visual projection area, is an approximate one -to -one spatial correspondence with the retina. Stimulation of nerve cells in this projection area by a weak electrical current causes the subject to see elementary visual events, such as coloured spots or flashes of light, in the expected location of the visual field.

In the human, the region of the striate cortex immediately surrounding the visual projection area is called the visual associate area. Electrical stimulation of cells in the association area gives rise to complex recognizable visual hallucinations (images of known objects or even meaningful action sequences).

Appendix B

SIGNAL THEORY

"Mere learning, mere humanitarianism, divorced from actual experience, may spell disaster to the cause espoused".
– Mahatma Gandhi –

Signal in its broadest sense is a quantity which, in some manner, conveys information about the state of a physical system. We can think of a signal as a result of measurement on a physical system under observation. The measuring mechanism converts the raw data to a form that makes sense to the observer.

Definition B.1 (Signal) *The variation through time of any significant physical quantity, occurring in a useful device or system.*

Examples of signal would be - Variation in Air pressure in front of a microphone as a sound wave impinges on it, or the current delivered by an electric generator.

What needs to be noted is that a physical quantity need not be measured or observed in order to qualify as a signal.

The study of physical systems can be broken down into the study of its smallest components. And as per the definition above almost any variable in such a system may be regarded as a signal. The signal can be thought of as the input and output quantities of the system, as well as the intervening quantities occurring at the inputs and outputs of various functional units. To be precise, one should refer to the various signals at various points in a system, instead of one signal in the system, since the variation with time at various points in a system may be quite different [LM65].

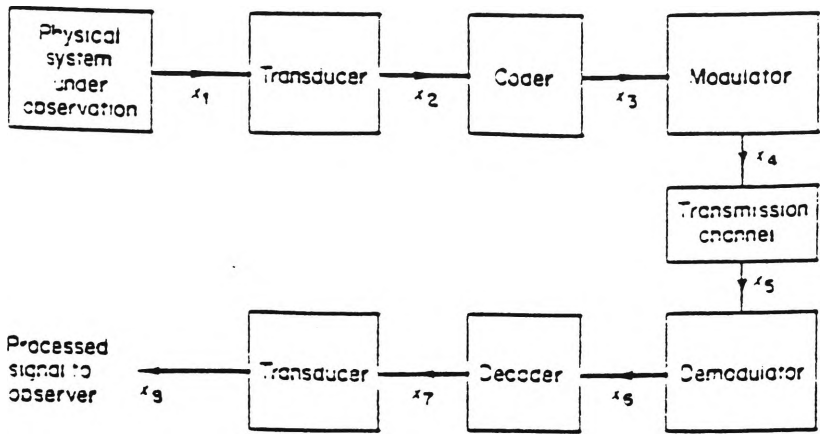

Figure B.1: Signal processing system.

Since a signal is regarded as a function of time representing a physical quantity, most of the properties attributed to it will be mathematical in nature.

Signals may be represented as a waveform with the variable under observation depicted in terms of variation with passing time. But for the purpose of analysis a signal may be represented as a mathematical function of time, or perhaps a group of functions in time, each defined over a specified time interval.

The above discussion points to a wide variety of signal formats. A signal theory, if it is to deal with a wide spectrum of signals, should be sufficiently generalized. Signal theory should include the study of:

- Methods for analytical representation of signals.
- Study of numerically quantifiable signal characteristics.
- Properties of signal processing tools and devices.

For ease of human interpretation, a signal is quite often expressed as a graph. But this representation does not readily lend itself to machine processing. For machine processing of the signal data, it should be represented by a subset of the original signal such that, the original can be reconstructed from the subset. Taking a subset of the signal, such that the elements are values of the signal equally spaced in time, serves the purpose.

B.1 Signal space

In order to make the theory more generalized, so that it covers a broader class of signals, one can introduce the concept of *signal spaces*. A signal may be considered as

a single point in space, or a single element of a set. This set of signals can be defined, so as to satisfy a property P, such that it includes as wide a range of useful signals. The property P should be sufficiently restrictive that it limits the class of signals included in the set small enough to be manageable, but at the same time it should not exclude many signals with interesting properties. In that sense the choice of property P is dependent on the signal processing application at hand.

Some of the classes of signals that are of interest to us are:

- Sinusoidal signals.

  If A denotes the set of all sinusoidal signals then

$$A = x; x(t) = Re[e^{\alpha + j(\theta + 2\pi ft)}], -\infty < t < \infty, \ \ \alpha, \theta, f \in R$$

- Periodic signals.

  If B is the set of periodic signals, with period T, then

$$B = x; xt + T = x(t), -\infty < t < \infty$$

- Bounded signals.

  Let C be the set of values bounded by a real positive number K.

$$C = x; |x(t)| \leq K, -\infty < t < \infty$$

- Energy-bounded signals.

  For

$$D = \left\{ x; \int_{-\infty}^{\infty} x^2(t)dt \leq K \right\}$$

  D is said to be the energy content of a signal. This interpretation of D may be attributed to the fact that if $x(t)$ is the voltage across a 1Ω resistor, the square of the voltage integrated over time is the energy dissipated by the load [EF69].

- Band limited signals.

  If E is the set of signals with frequencies less than U, E maybe expressed as :

$$E = \left\{ x; X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}\partial t = 0 \forall |f| > U \right\}.$$

The primary interest is in the Energy Bounded signals $f(x) \in L^2(R)$ such that

- For a pair of functions $f(x) \in L^2(R)$ and $g(x) \in L^2(R)$, the inner product of $f(x)$ and $g(x)$ is given by

$$< g(x).f(x) >= \int_{-\infty}^{+\infty} g(x)\overline{f(x)}.dx$$

where $\overline{f(x)}$ is the complex conjugate of $f(x)$.

- The *norm* of $f(x)$ in $L^2(R)$ is given by

$$||f||^2 = \int_{-\infty}^{+\infty} |f(x)|^2 dx$$

- The convolution of the two functions $f(x) \in L^2(R)$ and $g(x) \in L^2(R)$ is given by

$$f * g(x_0) = \int_{-\infty}^{+\infty} f(x)g(x_0 - x)dx$$

- The dilation of a function $f(x) \in L^2(R)$ by a scaling factor s is denoted as

$$f_s(x) = \sqrt{s}f(sx)$$

- The mirror image function of $f(x)$ about the origin is given as

$$\tilde{f}(x) = f(-x)$$

- And the Fourier transform of the function $f(x) \in L^2(R)$ is given by

$$\hat{f}(u) = \int_{-\infty}^{+\infty} f(x)\epsilon^{-\imath ux}dx$$

The class of functions mentioned above is integrable on the $L^2(R)$ space i.e. the integral converges to a finite value. Rigorous mathematical proof for the above statements for one dimensional signals is possible, but proving the extension of the results to the two dimensional case i.e., proving the above requirements in order to develop a set of reliable techniques for the purpose of signal processing in the $L^2(R)$ space, is somewhat difficult [VN92], [EF69], [Pap68].

Appendix C

IMAGE MANIPULATION

"Enjoy the world in a disinterested way".
– Ishopanishad –

During the course of this project a lot of activity had to be undertaken which had an indirect bearing on the successful completion of this project. One of the most important aspect was that of developing a library of utilities that would enable convenient and speedy handling of different image formats. The following is a graphic representation of the image data path as it was converted from one format to another. The routines that were incorporated in the development of this library and image manipulation are shown in the figure C.1 and the figure C.2.

The *pbmplus* library of functions that is widely available as a public domain software

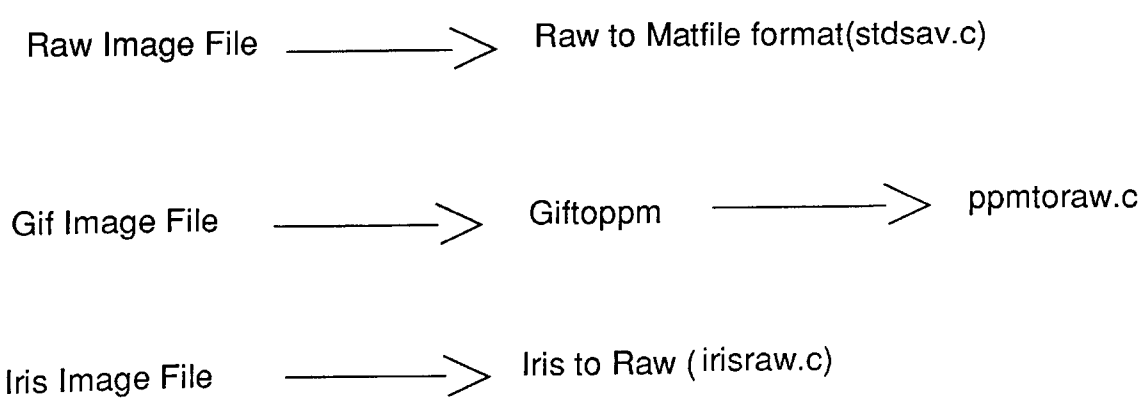

Figure C.1: This set of routines, converts the available image data to matfile format.

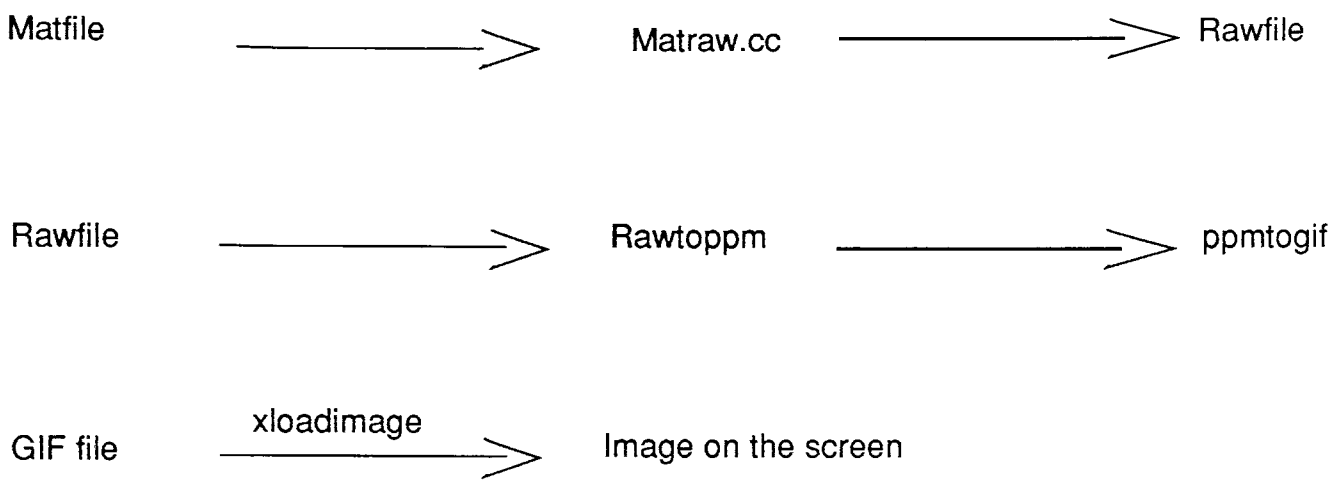

Figure C.2: This set of routines converts the matlab output to a format that may be displayed.

enables the conversion of image data files from $gif \Longrightarrow ppm \Longrightarrow pgm \Longrightarrow ps$ format, but it is difficult to get into this library if one has image data that is essentially, a flat file or a binary file. Since all the images that were grabbed for this particular project were essentially binary files, routines had to be developed to convert binary files to more amenable forms like ppm so as to enter the pbmplus function library.
Some of the routines developed for this project which make this collection of software a library that may be used as a general purpose development system for image processing applications as well as any signal processing application, are:

- Raw image file format conversion to three colour Gray scale image. i.e. rawtogray.c, which could then be used for conversion to PPM format using the *pbmplus* library calls.
- some of the images were acquired from the iris system frame grabber - so a routine for conversion from Iris format image to raw binary file was developed. i.e. irisraw.c
- Image files that are to be processed using the Matlab signal processing tool box have to be converted to the matfile format. A routine for the conversion directly from compressed raw files to matfile format was developed. i.e. stdsav.c
- Those images that were acquired in the gif format were converted into ppm using the *pbmplus* library call, but to convert the ppm file to raw file and thence to matfile format–a routine for conversion from ppm to raw was developed. i.e. ppmtoraw.c; This routine is useful in converting any "*rgb*" file(colour image) to a black and white(grayscale image).

- A routine to convert raw files to sun raster files was developed which was useful at times when working with utilities provided by Sun Microsystems i.e. if one wants to use the snapshot utility.
- Matlab m-files for various applications ranging from Gabor filter function mask generation, finding the non-negative peaks in a two dimensional function, an m-file to find out the exact frequency which is predominant in a given image function.
- Some UNIX script files to convert matlab files and display them in one smooth operation were also developed.

The exact utilization of the above routines and the path along which the image data was manipulated to achieve the desired results is depicted graphically in the figure C.1 and figure C.2
The following are the listings of source code developed over a period of one year for this project.

```
/*
/*  This program converts a matfile to a raw binary file.
/*  matraw.c
/*

#include <signal.h>
#include <stdio.h>
#include <string.h>
#define   image_siz     300*300


typedef struct Fmatrix{
        long type;   /* type */
        long mrows;  /* row dimension */
        long ncols;  /* column dimension */
        long imagf;  /* flag indicating imag part */
        long namlen; /* name length (including NULL) */
} Fmatrix;

FILE    *fp, *infile, *fp1;
```

```
int     imagf,i;
char    filename[80];
unsigned char image3[image_siz], dummy = 1.0;
double  rawim[image_siz];       /* pointer to raw image data */
double  imagim[image_siz];      /* pointer to imag image data */


main()



{

        int type;       /* Type flag: Normally 0 for PC, 1000 for Sun, Mac, and
        /* Apollo, 2000 for VAX D-float, 3000 for VAX G-float    */
        /* Add 1 for text variables.                            */
        /* See LOAD in reference section of guide for more info.*/
        int mrows;      /* row dimension */
        int ncols;      /* column dimension */
        /* int imagf;                   imaginary flag */
        char pname[10];                 /* pointer to matrix name */
        Fmatrix x;
        long namelen;


        printf("input filename\n");
        /*      input filename          */
        scanf("%s", filename);
        if ( (infile = fopen(filename,"r")) == NULL) {
                printf ("can't open %s\n",filename);
                exit (1);
        }
        fread(&x, sizeof(Fmatrix), 1, infile);
        fread(pname, sizeof(char), x.namlen, infile);
        /*      printf("length of name string = \n", x.namelen);        */


        printf("O.K. Fmatrix & name read-in\n");
```

```
/*      if  ((rawim = (double*)calloc(image_siz, sizeof(double))) == NU
/*              printf("Error in memory allocation");
/*              exit(1);
/*      }

fread( rawim, sizeof(double), image_siz, infile);
if (imagf)
        fread(imagim, sizeof(double), image_siz, infile);
fclose(infile);

/*      if  ((image3 = (unsigned char*)calloc(image_siz, sizeof(unsigne
/*              printf("Error in memory allocation");   */
/*              exit(1);                                */
/*      }                                               */

for ( i = 0; i < image_siz; i++) {
        image3[i] = (unsigned char *)rawim[i];
}

if ( (fp1 = fopen("matraw.dat","w")) == NULL) {
        printf ("can't open %s\n",filename);
        exit (1);
}
printf ( "matraw.dat opened\n");
fwrite (image3, sizeof(unsigned char), image_siz, fp1);

fclose(fp1);
free(rawim);
if (imagf)
        free(imagim);

}
```

```
/*
/* This program converts a one monochrome image file to an {\em rgb}
/* image file
/* rawtogray.c
/*

#include        "include.h"
#include        <string.h>
#define image_siz       256*256
#define mrows           256
#define ncols           256




/*
 * #include     </usr/openwin/include/X11/X.h>
 * #include     </usr/openwin/include/X11/Xlib.h>
 * #include     </usr/openwin/include/X11/Xutil.h>
 *
 */

unsigned char   image[image_siz];

main()
{
        FILE            *fp, *infile;
        int             i, j, k;
        char            filename[80];

        /*
        printf("input filename\n");
        scanf("%s", filename);



         if ((infile = fopen(filename, "r")) == NULL) {
                printf("can't open %s\n", filename);
```

```
                exit(1);

        }
*/

        fread(image, sizeof(unsigned char), image_siz, stdin);

        /*
        fclose(infile);
*/


        if ((fp = fopen("im.col", "w")) == NULL) {

                printf("can't open im.col\n");
                exit(1);
        }



        for (i = 0; i < image_siz; i++) {


                fwrite(&image[i], sizeof(unsigned char), 1, fp);

                fwrite(&image[i], sizeof(unsigned char), 1, fp);

                fwrite(&image[i], sizeof(unsigned char), 1, fp);

        }

        fclose(fp);

}
```

```
/*
/* ppmtoraw.c
/* This routine is very useful in converting an rgb colour image file
/* to a black & white (grayscale) image.

#include <signal.h>
#include <stdio.h>
#include <string.h>

typedef struct fmatrix{
        char magic[2];   /* magic number for ppm file */
        char nline1;  /* newline character */

        char mrows[3];  /* no. of rows */
        char wspace;  /* newline character*/

        char ncols[3]; /* no. of columns */
        char nline2; /* newline character */
        char maxval[3];  /* maximum intensity value of the pixels */
        char nline3; /* newline character */
} Fmatrix;


FILE    *fp, *infile;
int     i;
char    filename[80], outfile[80];
unsigned char *image, *trash, *buff, *raw, r, g, b;
int image_siz;


main()
{

        int mrows;      /* row dimension */
        int ncols;      /* column dimension */
        int maxval;     /* maximum intensity value of the pixels */
        int h=100, m1, m2, m3;
```

```
int t=10, n1, n2, n3, v1, v2, v3, nline3;
char pname[10];     /* pointer to matrix name */
Fmatrix x;
/*        double  rawim[image_siz];      pointer to raw image data */
/*        double  imagim[image_siz];     pointer to imag image data */


printf("input filename\n");
scanf("%s", filename);

if ( (infile = fopen(filename,"rb")) == NULL)  {
        printf ("can't open %s\n",filename);
        exit (1);
}


if  ((trash = (unsigned char*)calloc(512, sizeof(unsigned char))) == NU
        printf("Error in memory allocation");
        exit(1);
}

fread(&x, sizeof(Fmatrix), 1, infile);

printf("%c,%c \n", x.magic[0],x.magic[1]);

printf("%c,%c \n", x.mrows[0],x.mrows[1], x.mrows[2]);

printf("%c,%c,%c \n", x.ncols[0],x.ncols[1],x.ncols[2]);

printf("%c,%c,%c \n", x.maxval[0],x.maxval[1],x.maxval[2]);

m1 = (int)(x.mrows[0] - 48);
m2 = (int)(x.mrows[1] - 48);

m3 = (int)(x.mrows[2] - 48);
n1 = (int)(x.ncols[0] - 48);

n2 = (int)(x.ncols[1] - 48);
```

```
n3 = (int)(x.ncols[2] - 48);

v1 = (int)(x.maxval[0] - 48);
v2 = (int)(x.maxval[1] - 48);

v3 = (int)(x.maxval[2] - 48);
nline3 = (int)(x.nline3 - 48);

mrows = (h*m1 + t*m2 + m3);
ncols =(h*n1 + t*n2 + n3);

maxval = (h*v1 + t*v2 + v3);

printf("%u,%u,%u,%u \n", mrows, ncols, maxval, nline3);


image_siz = mrows * ncols;

if  ((image = (unsigned char*)calloc(image_siz, sizeof(unsigned char)))
        printf("Error in memory allocation");
        exit(1);
}

if  ((raw = (unsigned char*)calloc(image_siz, sizeof(unsigned char))) =
        printf("Error in memory allocation");
        exit(1);
}


if  ((buff = (unsigned char*)calloc( 3, sizeof(unsigned char))) == NULL
        printf("Error in memory allocation");
        exit(1);
}


/*          fread(&trash, sizeof(unsigned char), 1, infile);     */
```

```
for ( i = 1; i < image_siz; i++) {

        fread(&buff[1], sizeof(unsigned char), 1, infile);

        fread(&buff[2], sizeof(unsigned char), 1, infile);

        fread(&buff[3], sizeof(unsigned char), 1, infile);



        r = buff[1];

        g = buff[2];

        b = buff[3];

        /*      printf("%u, %u, %u \n", r, g, b);       */

        raw[i] = (((30*r) + (59*g) + (11*b))/100);

        /*      printf("%d, ", raw[i]);         */

}

printf("I am here now");

fclose(infile);

printf("input outfile\n");
scanf("%s", outfile);

strcat(outfile, ".raw\0");



if ((fp = fopen(outfile,"wb")) == NULL) {
```

```
                printf ("can't open outfile\n");
                exit (1);
        }


        fwrite(raw, sizeof(unsigned char), image_siz, fp);

        fclose(fp);



}
```

```
/*
/* irisraw.c
/*
/*

#include <stdio.h>
#include <string.h>
#include "include.h"
#define   image_siz     512*512



main()
{

        FILE    *fp, *infile;
        double  dummy = 1.0;
        int     imagf,i;
        char    filename[80], outfile[80];
        unsigned char *image, *trash;
```

```
printf("input filename\n");
scanf("%s", filename);

if ( (infile = fopen(filename,"r")) == NULL) {
        printf ("can't open %s\n",filename);
        exit (1);
}



if  ((trash = (unsigned char*)calloc(512, sizeof(unsigned char))) == NU
        printf("Error in memory allocation");
        exit(1);
}


fread(trash, sizeof(unsigned char), 512, infile);


if  ((image = (unsigned char*)calloc(image_siz, sizeof(unsigned char)))
        printf("Error in memory allocation");
        exit(1);
}


fread(image, sizeof(unsigned char), image_siz, infile);

fclose(infile);


printf("input outfile\n");
scanf("%s", outfile);

strcat(outfile, ".grey\0");
```

```
        if ((fp = fopen(outfile,"w")) == NULL)
        {
                printf ("can't open outfile\n");
                exit (1);
        }


        fwrite(image, sizeof(unsigned char), image_siz, fp);

        fclose(fp);

}

/*
/* The following Matlab routine finds the peak values in the
/* Fourier transformed image.
/*

H = fft2(I);
t=length(H);
u=fix((length(H))/2);
D1(1:1:u,1:1:u) = H(1:1:u,1:1:u);
%M1=[1/4 0 -1/4;0 0 0;-1/4 0 1/4];


%DL = conv2(M1,H1);
%D1(1:1:128,1:1:128) = DL(2:1:129,2:1:129);
C1 = zeros(length(D1));
B1 = zeros(length(D1));
n = length(D1);
m = n;
m
n
for l=1 :n,
for k=3 :m - 2,
if (((D1(k,l) - D1(k-1,l)) > 0 ) & ((D1(k+1,l) - D1(k,l)) < 0 )) & (((D1(k-1,l)
```

```
% & (((D1(k-2,l) - D1(k-3,l)) > 0 ) & ((D1(k+3,l) - D1(k+2,l)) < 0 )) &
% (((D1(k-3,l) - D1(k-4,l)) > 0 ) & ((D1(k+4,l) - D1(k+3,l)) < 0 ))

% & (((D1(k-4,l) - D1(k-5,l)) > 0 ) & ((D1(k+5,l) - D1(k+4,l)) < 0 ))

C1(k,l) = D1(k,l);
end
end
end

for k=1 :n,
for l=4 :m - 3,
if (((D1(k,l) - D1(k,l-1)) > 0 ) & ((D1(k,l+1) - D1(k,l)) < 0 )) & (((D1(k,l-1)

B1(k,l) = 1;
end
end
end

mesh(C1);
```

The above is a sampling of some of the routines developed.

Appendix D

DYSLEXIA - Due to VISUAL PROCESSING ABNORMALITIES

"Educate a man - you educate one individual, educate a woman - you educate an entire family!"
– Unknown –

Research workers in psychology engaged in the analysis of causes for Specific Reading Disability among children and adults have reported links between losses in spatial frequency sensitivity with losses in reading or reading related tasks [LM86].

It is also reported that in some multiple schlerosis patients with losses in frequency sensitivity, the letter recognition decreases when the component spatial frequencies match the range of sensitivity loss [LM86] [BW72].

In an investigation Bodis-Wollner [BW72] report that adult patients with cerebral lesions or tumours have problems in everyday vision even though they show normal acuity. These patients had altered sensitivity in only some spatial frequency channels. These patients when treated for spatial-frequency processing reported a return to normal everyday vision.

The above findings should be taken as a fairly direct evidence that low-level visual deficits like loss of sensitivity to certain spatial frequencies, can lead to reading disabilities [LM86].

With active support from Prof. William Lovegrove of the Department Of Psychology, University Of Wollongong, this project, besides the machine vision objective, will aim to analyse the suitability of the Spatial/Spatial-frequency approach. Especially it will aim at studying the Gabor filter operator for the purpose of helping Dyslexic persons with specific-reading disability due to visual deficit alone, with no other factor contributing to the reading disability, such as speech impairment or language processing problems.

D.1 Interpretation of Results in terms of visual processing abnormalities - Dyslexia

Specific reading disability is a broad term which encompasses reading disabilities arising from a number of sources, like speech impairment and language processing disability.

Considered herein are only those cases having reading disability due to visual processing abnormalities.

As per Prof. William Lovegrove and Prof. Mary Williams carrying out research in the field of reading disability due to *Dyslexia*, last ten years research has indicated links between losses in spatial frequency sensitivity with losses in reading or reading related tasks [LM86] [LM86].

Prof. Lovegrove and Prof. Williams in their recent publication state that those adults reporting reading disability due to visual deficit or visual processing abnormalities, actually display different response times in their visual systems from a normal person [ML92].

These visually disabled readers show an improvement in their response to incoming stimulus, especially text matter with a higher spatial-frequency content when the text matter being read is blurred on purpose [ML92].

This controlled blurring of text matter for improved reading ability suggests that the spatial-frequency of the text matter needs to be altered to suit the spatial-frequency sensitivity of the reading disabled person [BW72] and [LM86].

In such an application an operator like the *Wavelet* that belongs to conjoint spatial/spatial-frequency domain may be used in real-time to identify those areas in a given text matter that contain the spatial-frequencies to which the disabled reader is insensi-

tive. This region then may be blurred in a controlled manner to present the text at a spatial-frequency that is easily sensed by the visual system of the disabled reader. This application needs a lot of experimental work which is not considered under this project.

This controlled blurring and other methods for improving the response time to a colour stimulus are techniques that rely on the modification or alteration in the localised spatial-frequency of the text matter, to suit the spatial-frequency response characteristic of the reading disabled.

From the above discussion, and the fact that the human visual system and the vision system of some species of animals higher up in the evolution ladder manage to reduce the quantity of data that needs to be processed in order to extract information about the suroundings, by way of reducing the sampling rate depending upon the distance of the object from the human eye, it can be inferred that the reduction in sampling rate mentioned here is a natural recourse for the human visual system when the incoming data rate becomes overwhelmingly large, i.e the spatial-frequency of the data becomes very high.

In terms of *Dyslexics* the spatial-frequency to which they are sensitive may be identified, and the text matter blurred to such a degree that the spatial-frequency content of the text being read is one of those to which the person is sensitive.

Bibliography

[AGM89] R. Cronland-Martinet A. Grossmann and J. Morlet. *Reading and Understanding Continuous Wavelet Transforms in Proceedings of the International Conference.* Springer, 1989.

[Bea90] Bovik and Clarke et. al. Multichannel Texture Analysis Using Localized Spatial Filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 12*, (1):55–73, 1990.

[Bro86] Arthur Browne. *Vision and Information Processing For Automation / Arthur Browne and Leonard Norton-Wayne.* Plenum Press, New York, 1986.

[BW72] I. Bodis-Wollner. Visual Acuity and Contrast Sensitivity in Patients with Cerebral Lesions. *Science*, (178):769–771, 1972.

[CGW87] Rafael C. Gonzalez and Paul Wintz. *Digital Image Processing.* Addison Wesley - Second Edition, Reading Mass., 1987.

[CS89] Jorge L. C. Sanz. *Advances in machine vision.* Springer-Verlag, c1989, New York., 1989.

[EF69] Lewis E. Franks. *Signal theory.* Prentice-Hall, Englewood Cliffs, N.J., 1969.

[Fol86] G. Folland. *Harmonic Analysis in phase space.* Princeton university press, New York., 1986.

[Gab46] D. Gabor. Theory of Communication. *J.I.E.E. vol. 93*, pages 429–457, 1946.

[GD80] John G. Daugman. Two-dimensional Spectral Analysis of Cortical Receptive Field Profiles. *Vision Research*, (20):847–856, 1980.

[GD85] John G. Daugman. Uncertainty Relation for Resolution in Space, Spatial Frequency, and Orientation Optimized By Two Dimensional Cortical Filters. *Journal of the Optical Society*, (A2):1160–1169, 1985.

[GD88] John G. Daugman. Complete Discrete 2-D Gabor Transforms by Neural Networks for Image Analysis and Compression. *IEEE Transactions on Acoustics, Speech and Signal Processing. Vol. 36*, (7):1169–1179, 1988.

[GM84] A. Grossmann and J. Morlet. Decomposition of Hardy Functions into Square Integrable Wavelets of Constant Shape. *SIAM Journal of Mathematical Analysis*, (15):723–736, 1984.

[GM89] Stephane G. Mallat. Multifrequency Channel Decompositions of Images and Wavelet Models. *IEEE Transactions on Acoustics, Speech and Signal Processing Vol. 37*, (12):2091–2105, 1989.

[GN89] H. G. Newman. *Industrial robotics, Machine Vision and Artificial Intelligence.* Howard W. Sams, Indianapolis, Ind., 1989.

[HCW92] Tao-I Hsu, A D Calway, and R Wilson. Analysis of Structured Texture Using the Multiresolution Fourier Transform. Technical report, Department of Computer Science, University of Warwick, Coventry CV4 7AL UK., 1992.

[HJ90] B.A. Harrison and D.L.B. Jupp. *Introduction to Image Processing.* CSIRO, Australia, Division of Water resources, 1990.

[Jan81] A.J.E.M. Janssen. Gabor Representation of Generalized Functions. *Journal of Mathematical Analysis And Applications vol. 83*, (1):377–394, 1981.

[JC85] R. J. Clarke. *Transform Coding Of Images.* Academic Press., London., 1985.

[JHSD73] R. J. Haralick, K. Shanmugam, and I. Dinstein. Textural Features for Image Classification. *Computer Methods in Image Analysis, IEEE Transactions on System, Man and Cybernetics Vol. SMC-3*, (6):610–621, 1973.

[JS89] R. J. Schalkoff. *Digital Image Processing and Computer Vision.* Wiley., New York., 1989.

[KPH86] Berthold Klaus Paul Horn. *Robot vision.* Cambridge, Mass: MIT Press; New York: McGraw-Hill, c1986., New York., 1986.

[LC82] James L. Crowley. A Representation for Visual Information. Technical Report CMU-RI-TR-82-7, Carnegie-Mellon University, The Robotics Institute, 1982.

[LM65] James L. Marshall. *Introduction to signal theory.* International Textbook Company, Scranton, Pa, 1965.

[LM86] William Lovegrove and Frances Martin. A Theoretical and Experimental Case for a Visual Deficit in Specific Reading Disability. *Cognitive Neuropsychology*, (3):225–267, 1986.

[LYU90] A. L. Yuille and S Ullman. *Visual Cognition and Action.* MIT press - Editors Koslyn et. al, Mass., 1990.

[MH83] R. et al M. Haralick. *Fundamentals in Computer Vision.* University Press, Cambridge, New York., 1983.

[ML92] C. Williams Mary and William Lovegrove. *Sensory and Perceptual Processing in Reading Disability.* Elsevier Science Publishers B.V., New York, N.Y., U.S.A., 1992.

[Nag90] G. Naghdy. Multisensory Perception for Mobile Robots in a Message Passing Environment. In *International Conference on Automation Robotics and Computer Vision (ICARVC90)*, Singapore, September 1990.

[Nib86] Wayne Niblack. *An Introduction to Digital Image Processing.* Prentice/Hall International, 1986.

[Pap68] Athanasios Papoulis. *Systems and Transforms with Applications in Optics.* McGraw-Hill Book Company, New York., 1968.

[P:G82] *Hierarchical Processing of Structural Information in Artificial Intelligence.*, Paris, 1982. IEEE International Conference on Acoustics, Speech and Signal Processing.

[Phi89] Philips. *Philips Camera Manual.* Philips, Eindhoven, Netherlands., 1989.

[Pre91] William H. Press. Wavelet Transforms. *Harvard-Smithsonian Center for Astrophysics Preprint*, (3184):1–14, 1991.

[PYZ88] Moshe Porat and Yehoshua Y. Zeevi. The Generalized Gabor Scheme of Image Representation in Biological and Machine Vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 10*, (4):452–467, 1988.

[Ran91] Surendra Rangnath. Image Filtering Using Multiresolution Representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 13*, (5):426–440, 1991.

[RRW90] Todd R. Reed and Harry Wechsler. Segmentation of Textured Images and Gestalt Organisation Using Spatial/Spatial-Frequency Representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 12*, (1):1–12, 1990.

[RV91] Olivier Rioul and Martin Vetterli. Wavelets and Signal Processing. *IEEE Signal Processing Magazine*, (1):14–38, 1991.

[Tur86] M. Turner. Texture Discrimination by Gabor Functions. *Biological Cybernetics*, (55):71–82, 1986.

[VN92] R. V. Nillsen. *Lecture Notes in Measure & Integration.* University Of Wollongong, Dept. Of Mathematics, Wollongong, 1992.

[WADCD92] R Wilson, E R S Pearson A D Calway, and A R Davies. An Introduction to the Multiresolution Fourier Transform. Technical report, Department of Computer Science, University of Warwick, Coventry CV4 7AL UK., 1992.

[WCG77] H. Wilson and S. C. Giese. Threshold Visibility of Frequency Gradient patterns. *Vision Research Vol. 17*, (1):1177–1190, 1977.

[WCP92] R Wilson, A D Calway, and E R S Pearson. A Generalised Wavelet Transform for Fourier Analysis: the Multiresolution Fourier Transform and its Application to Image and Audio Signal Analysis. *IEEE Trans. on Information Theory*, 38(2):674–690, March 1992.